



Minería de datos y Patrones

Version 2024-I

SVM

Dr. José Ramón Iglesias
DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingeniería Electrónica
Universidad Popular del Cesar

Objetivos

- Identificar el método de funcionamiento de una máquina de soporte vectorial.
- Reconocer el principio de funcionamiento del truco del kernel.
- Usar una SVM en el problema de clasificación binaria.

Que son:

Son un conjunto de métodos de aprendizaje supervisado para la clasificación y regresión.

Viendo los datos de entrada como conjuntos de vectores en un espacio n -dimensional, una SVM construirá un hiper plano de separación en ese espacio, que maximiza el margen entre los conjuntos de datos.

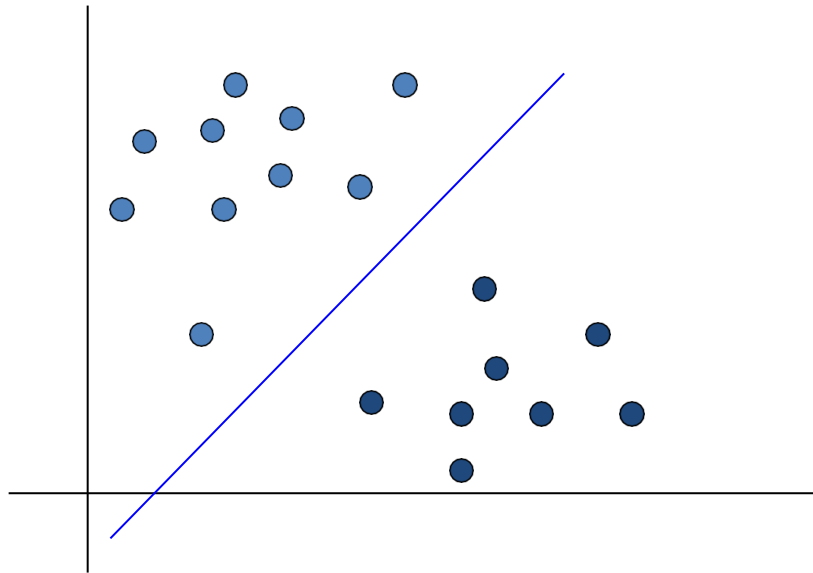
Se hicieron populares por su éxito en reconocimiento de dígitos manuscritos

Historia:

https://scholar.google.com/citations?user=U_IVY50AAAJ&hl=en

<https://scholar.google.com/citations?user=vtegaJgAAAJ&hl=en>

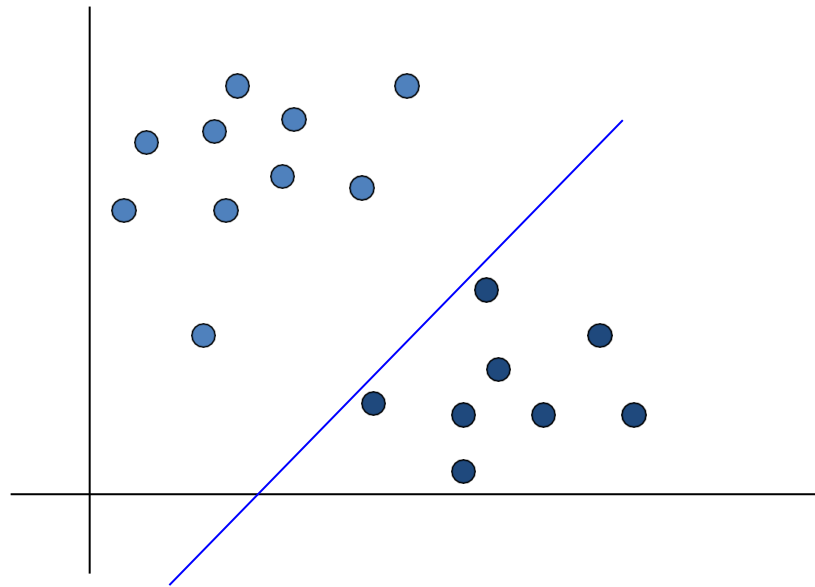
Idea:



Por ejemplo:

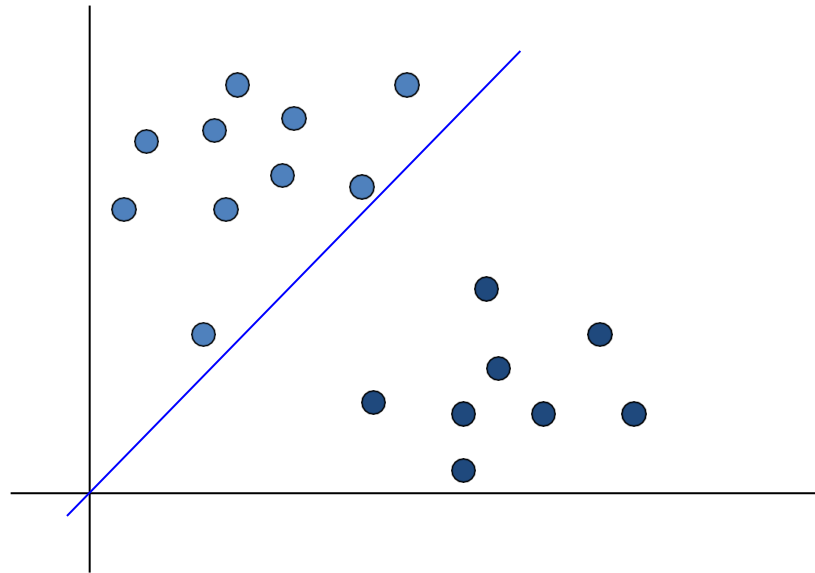
Separar en este
caso dos clases
con una línea
recta

Idea:



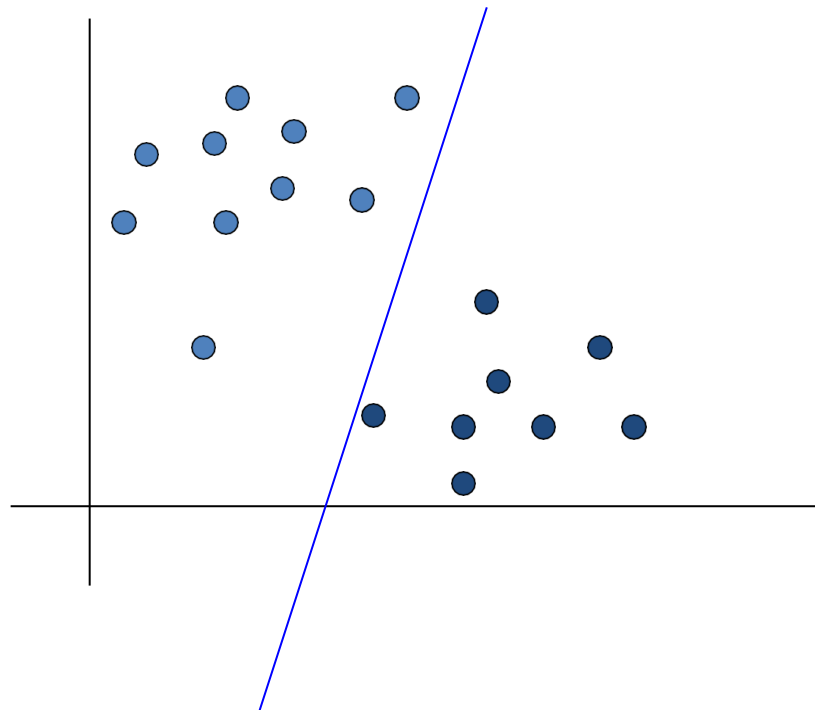
Se puede
así

Idea:



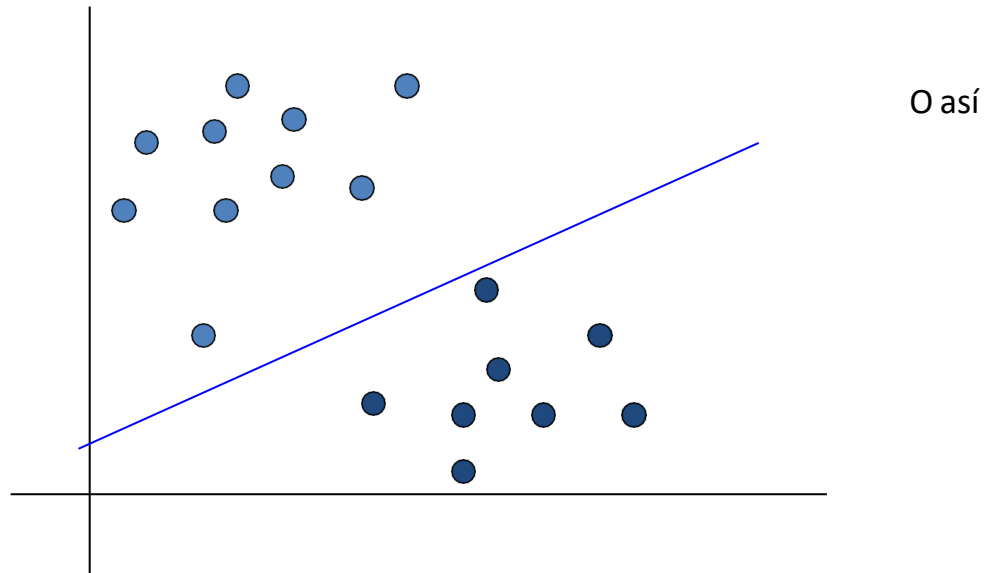
Se puede
poner
también así

Idea:

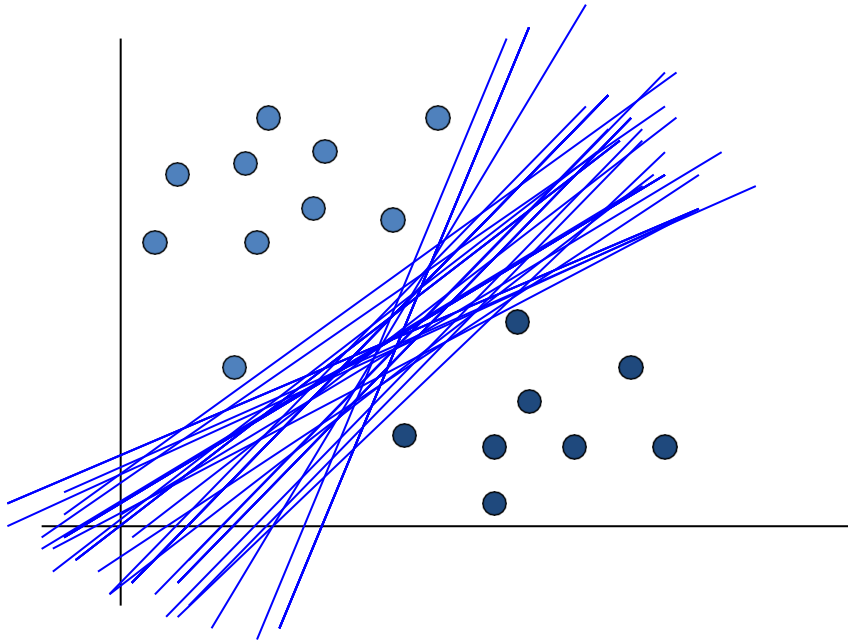


O así

Idea:

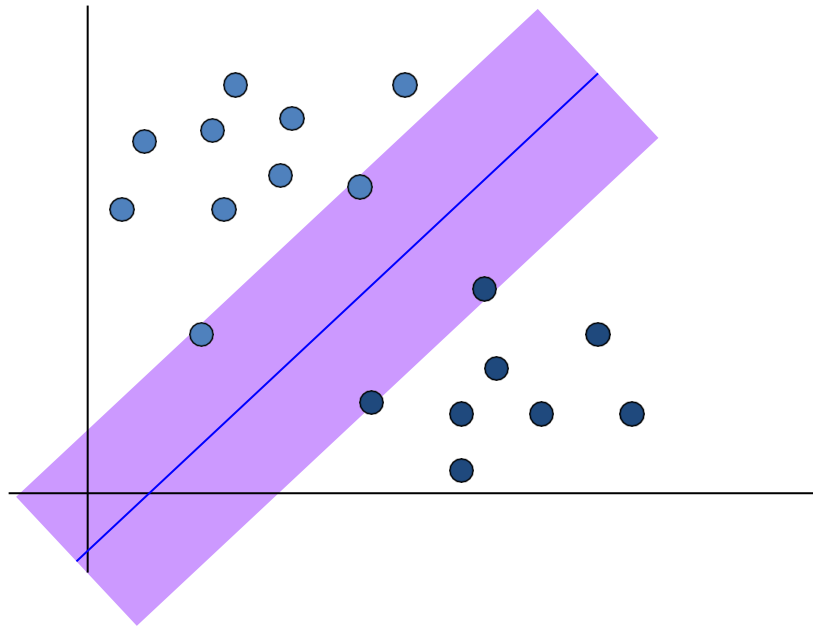


Idea:



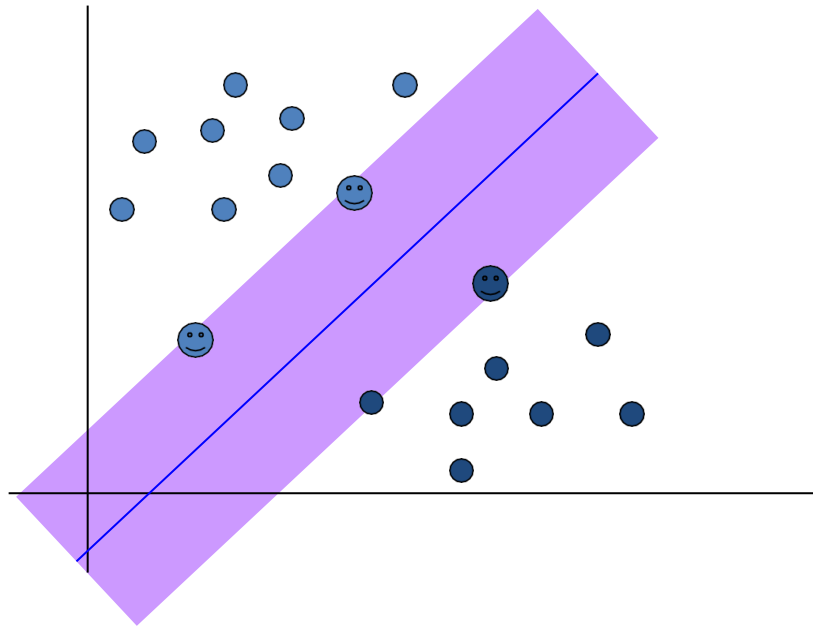
?

Idea:



La línea que
tenga un mayor
margen será la
escogida para
clasificar

Idea:

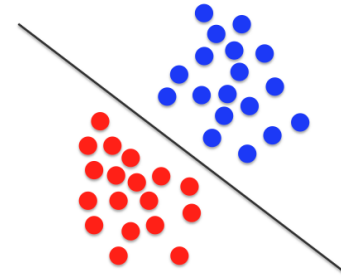


Los limites que
toca este margen
son llamados:

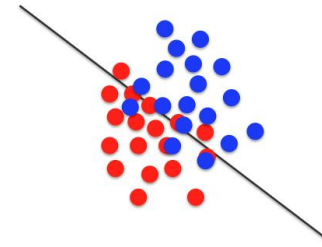
**VECTORES DE
SOPORTE**

SVM: Máquinas vectoriales de soporte

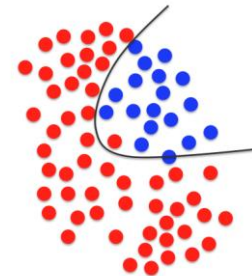
1) Lineal con separación perfecta



2) Lineal sin separación perfecta

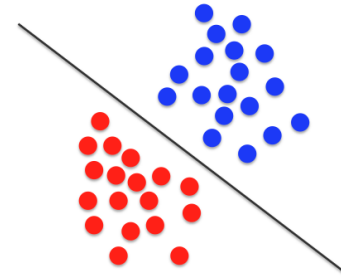


3) No lineal

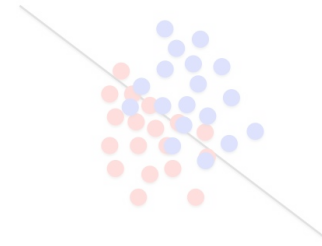


SVM: Máquinas vectoriales de soporte

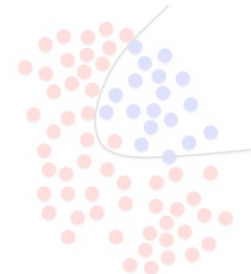
1) Lineal con separación perfecta



2) Lineal sin separación perfecta



3) No lineal



Maquinas de soporte vectorial lineales

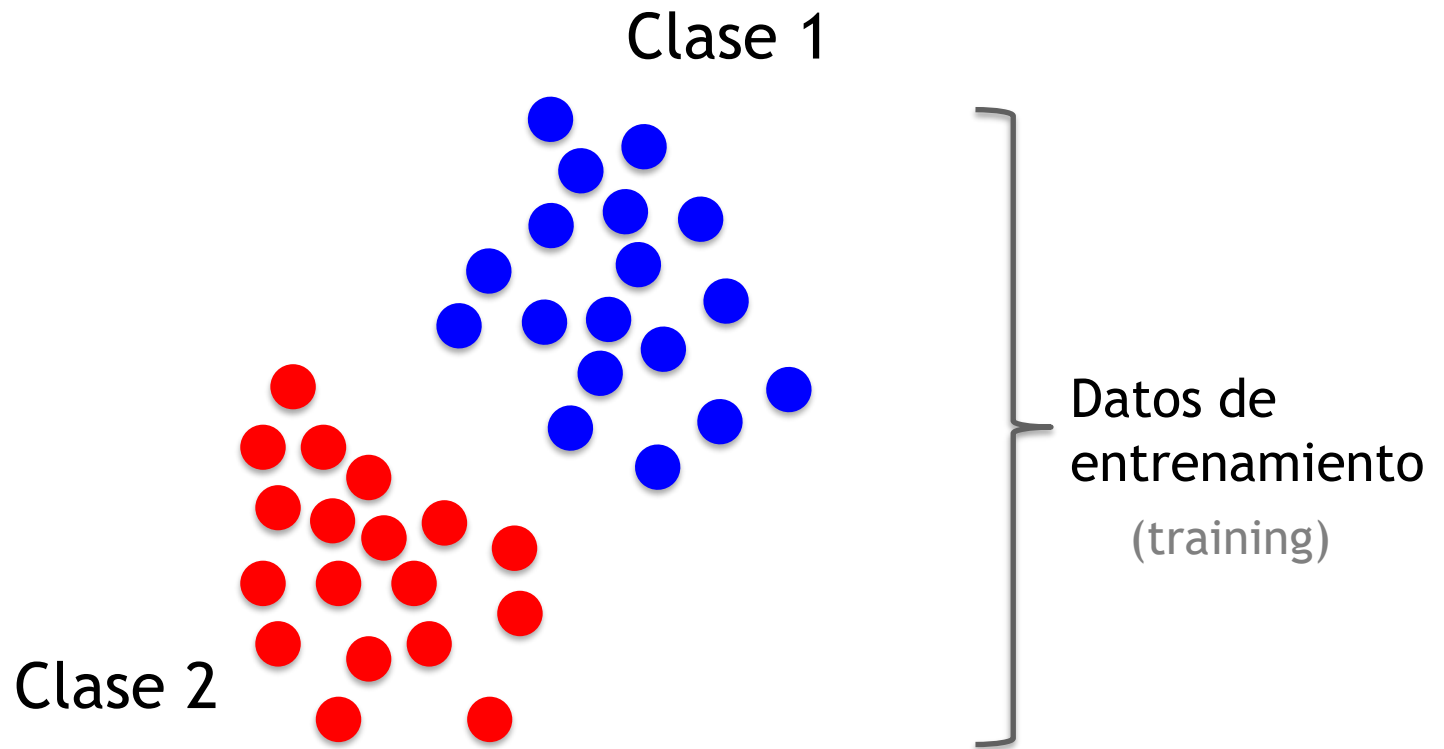
Caso separable

Dadas m observaciones:

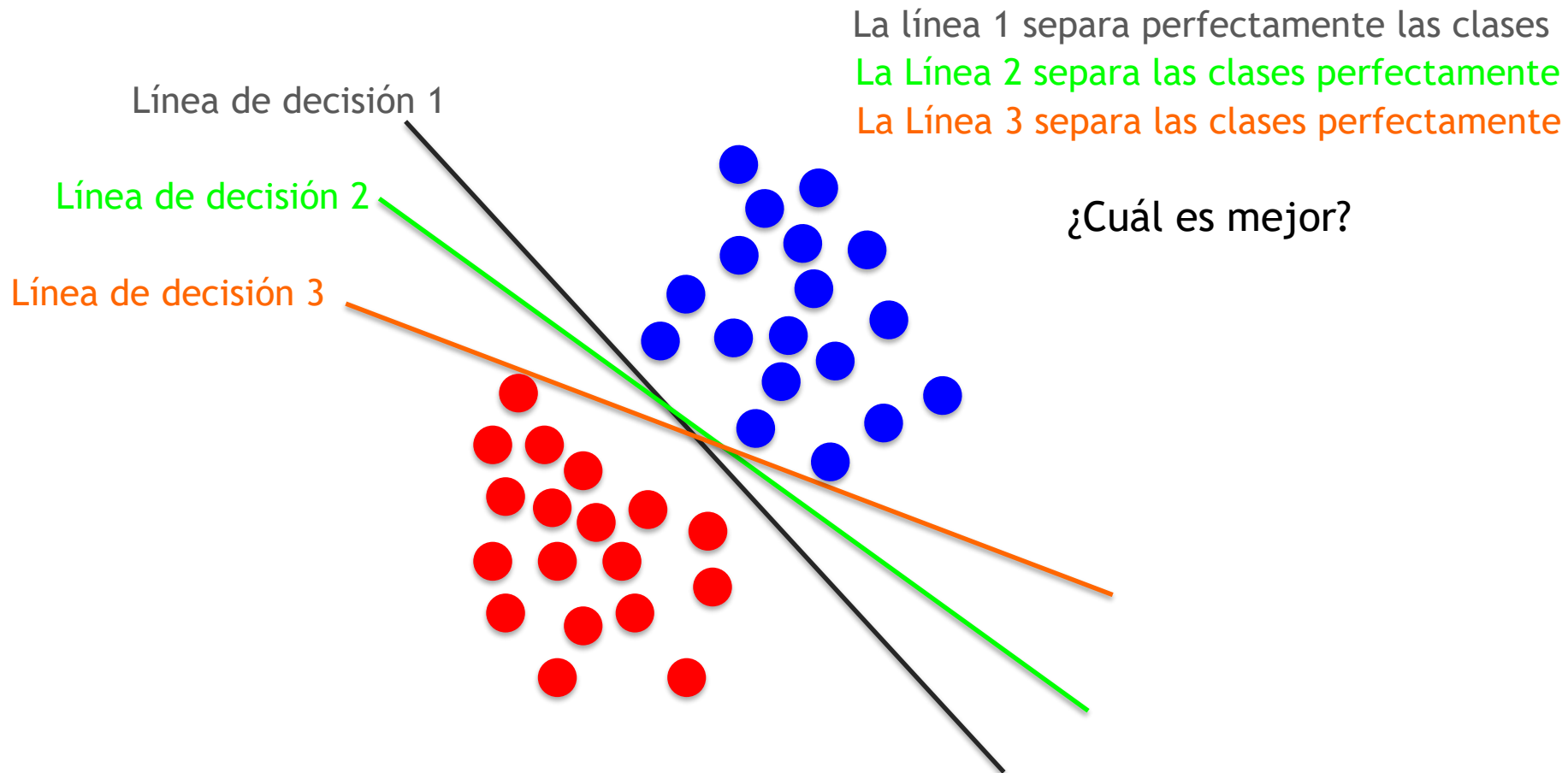
Con un vector de características $x_i \in R^n, i = \{0 \dots m - 1\}$

Y unas etiquetas $y_i \in \{+1, -1\}$

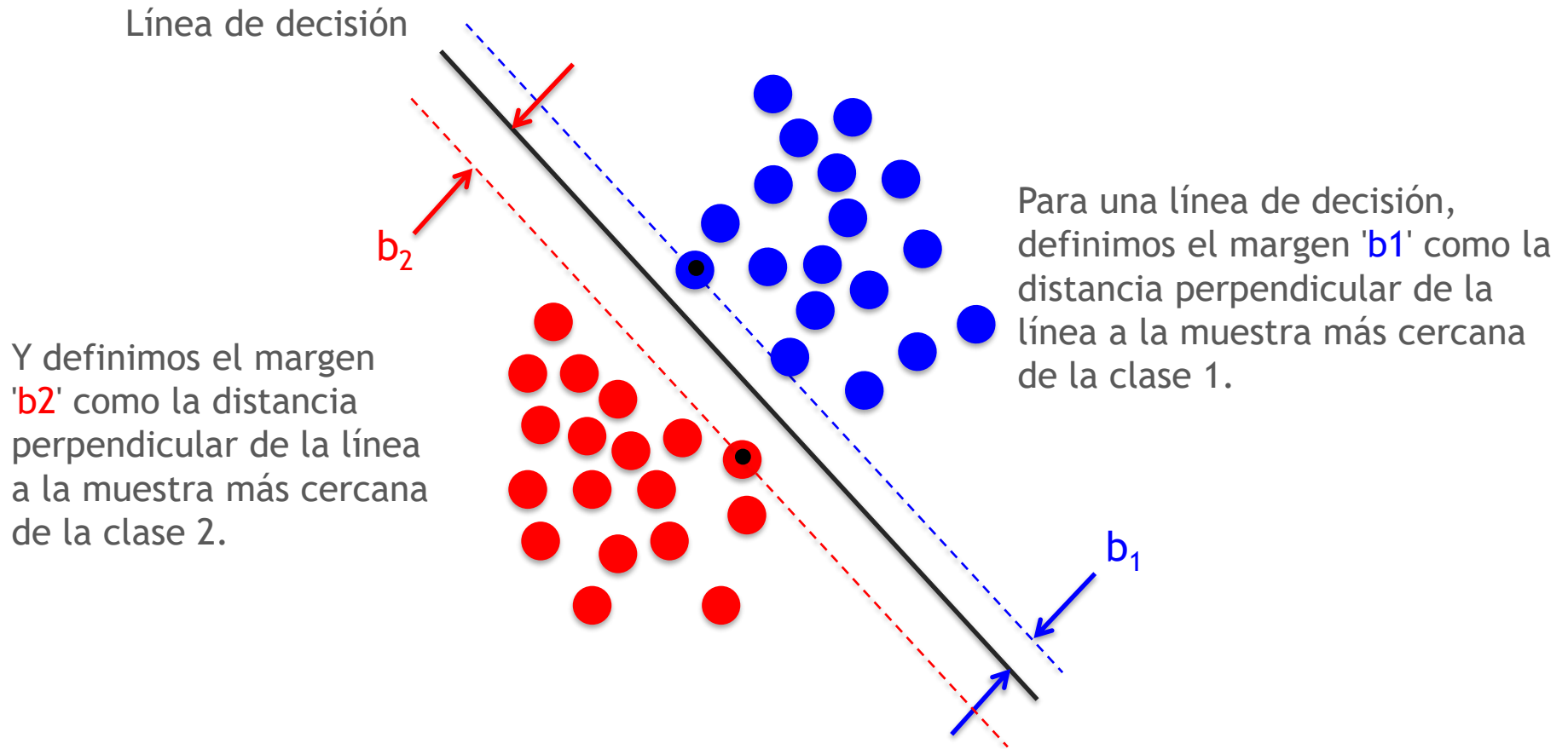
SVM: dos clases



SVM: dos clases



SVM: dos clases

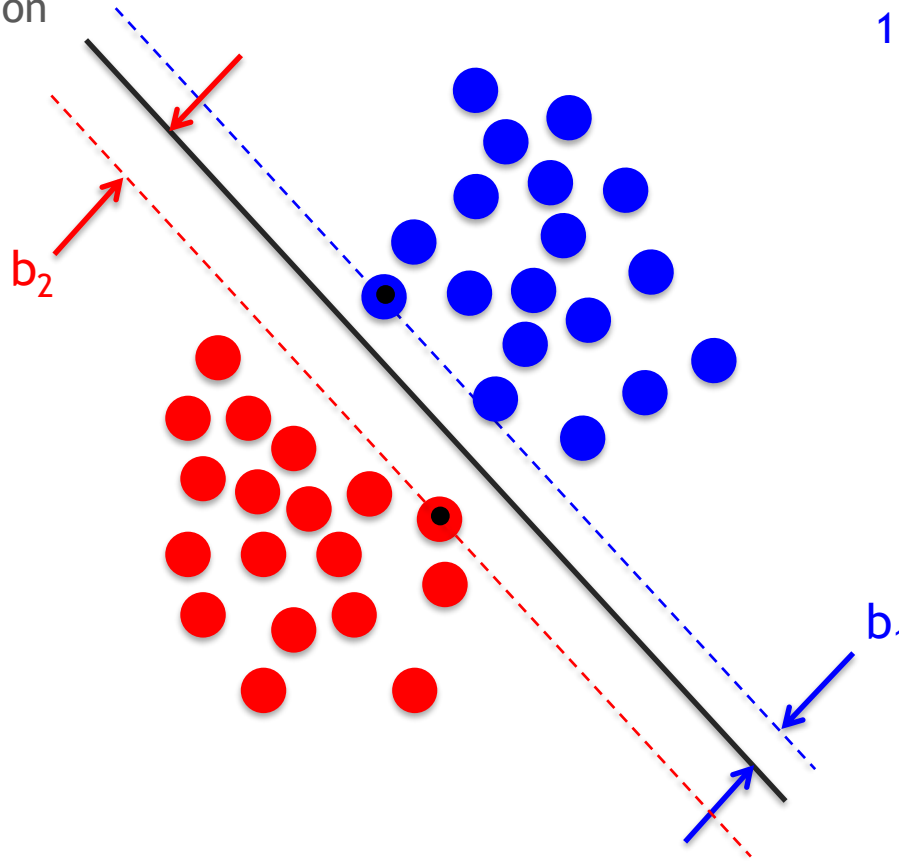


SVM: dos clases

Ideas clave de SVM:

1) $b_1 = b_2 = b$.

Línea de decisión

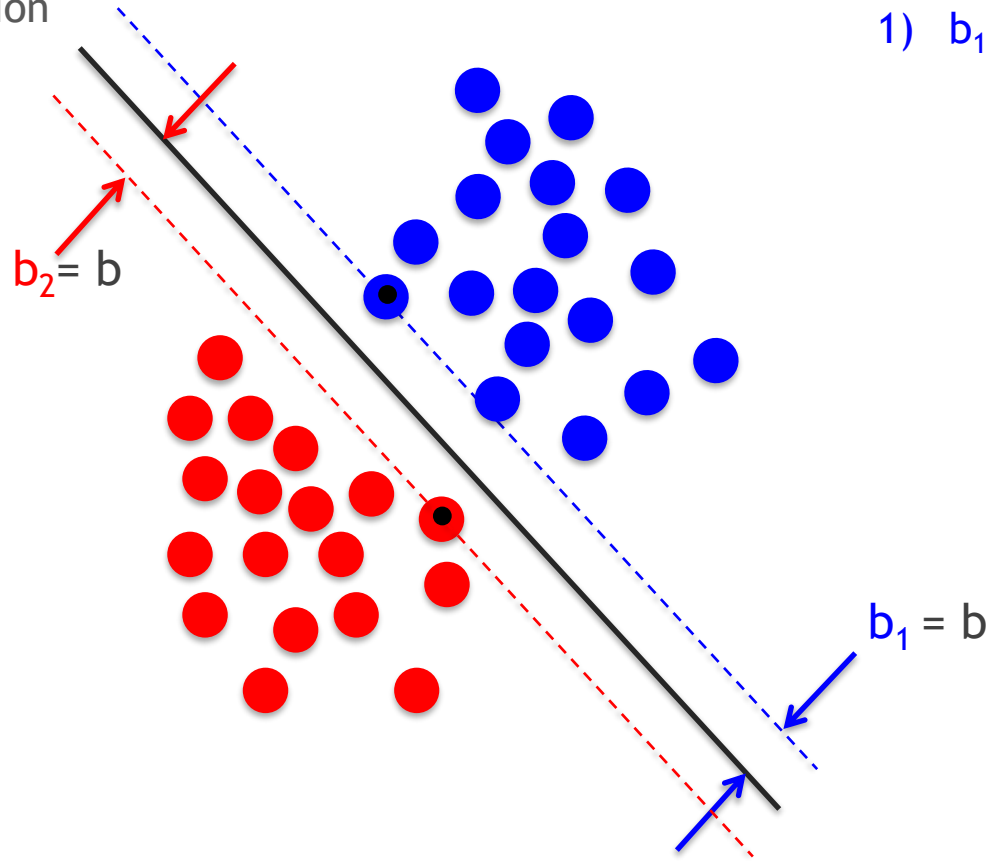


SVM: dos clases

Ideas clave de SVM:

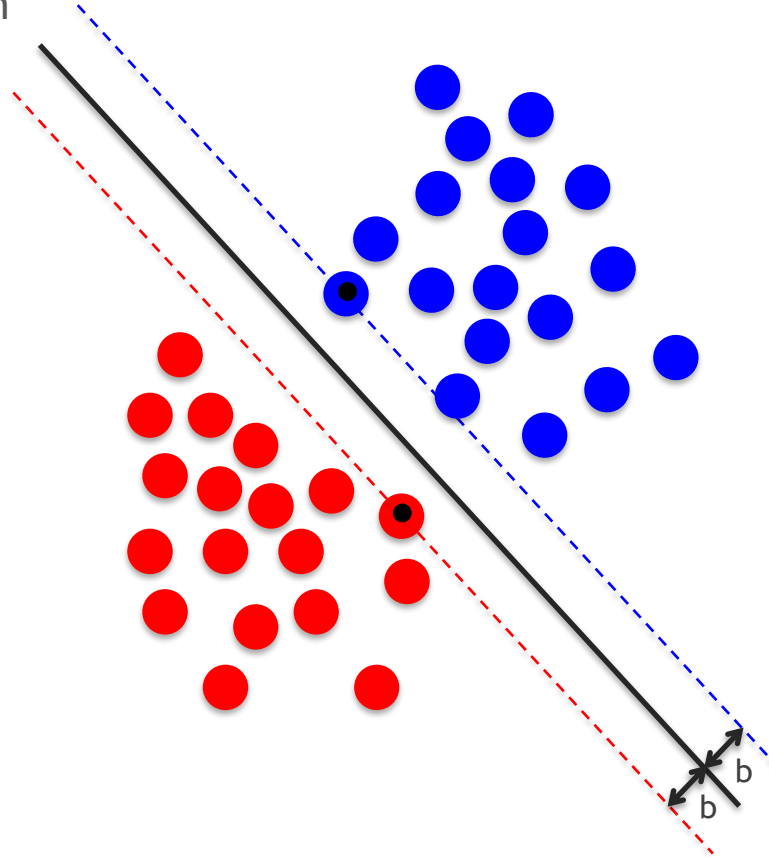
1) $b_1 = b_2 = b$.

Línea de decisión



SVM: dos clases

Línea de decisión

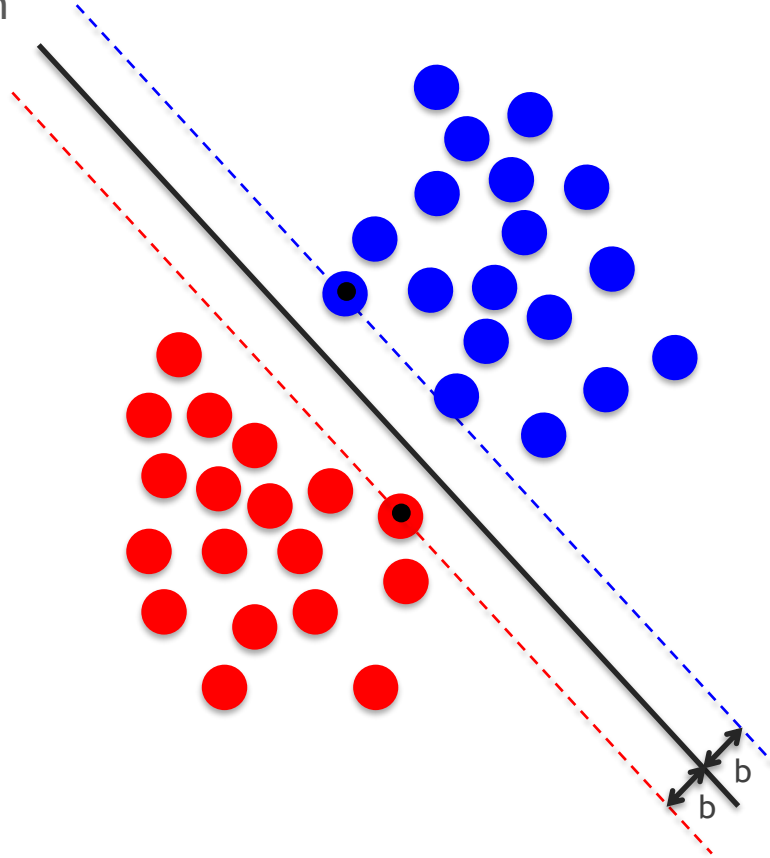


Ideas clave de SVM:

1) $b_1 = b_2 = b$.

SVM: dos clases

Línea de decisión



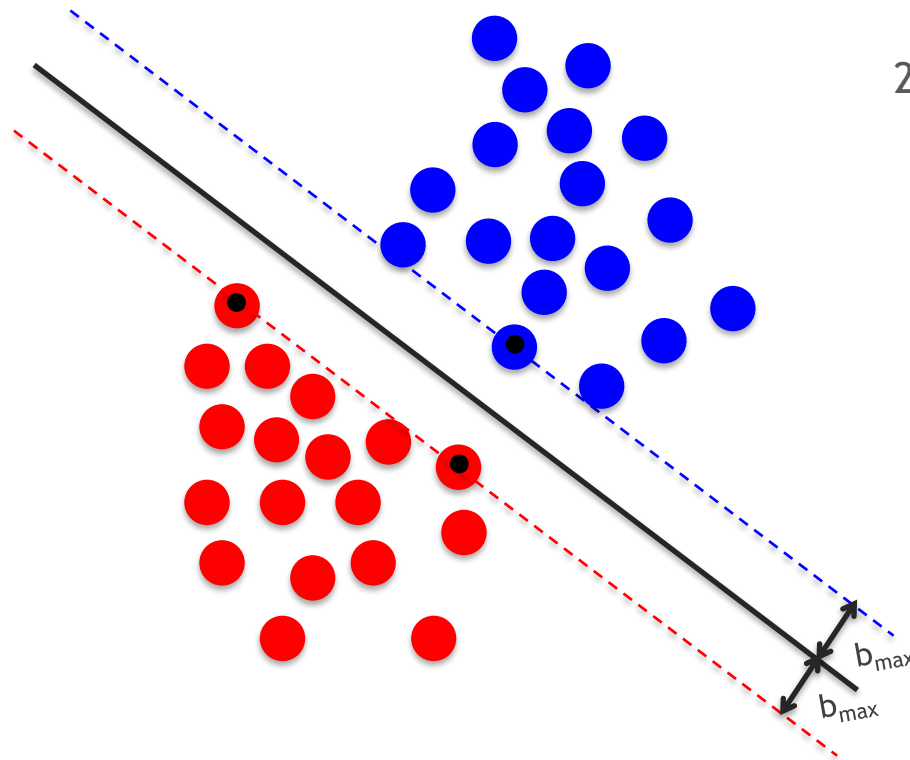
Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.

SVM: dos clases

Línea de decisión



Ideas clave de SVM:

1) $b_1 = b_2 = b$.

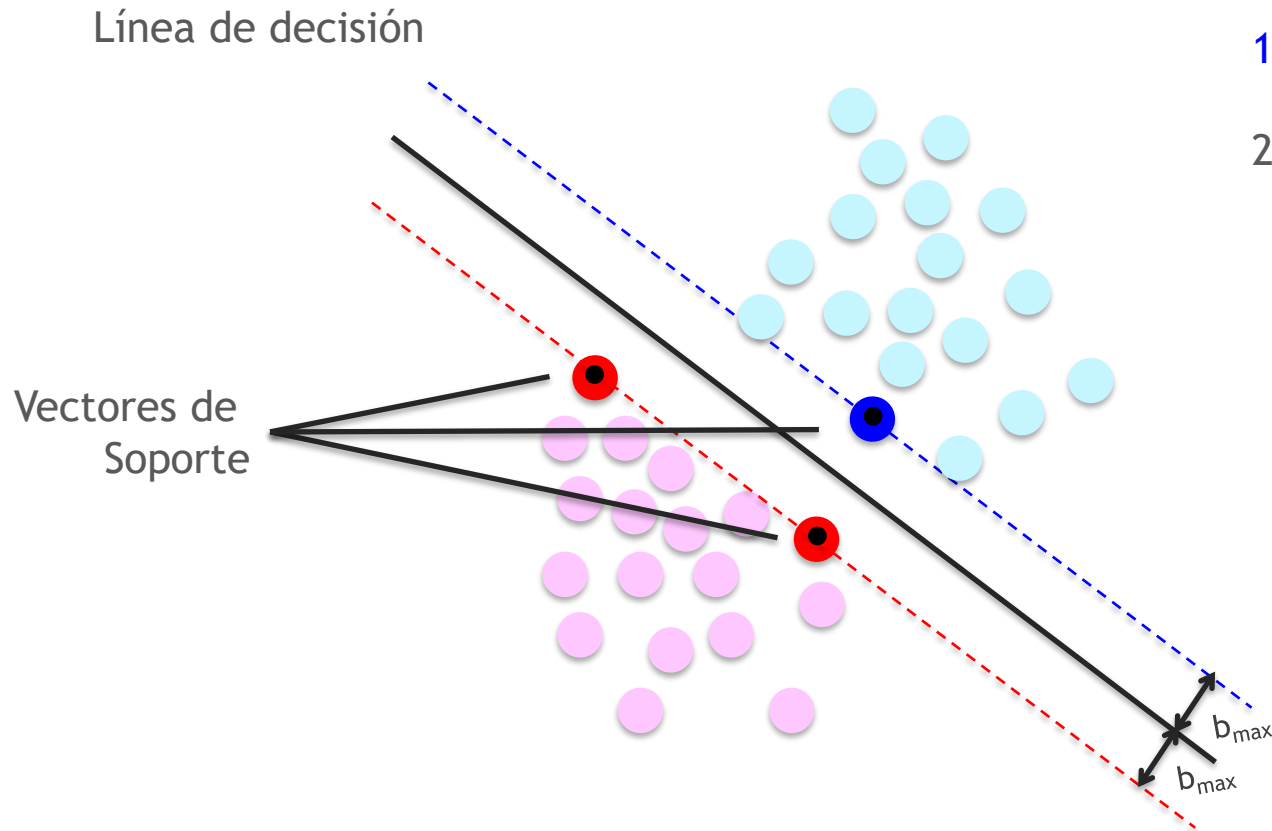
2) b debe maximizarse.

SVM: dos clases

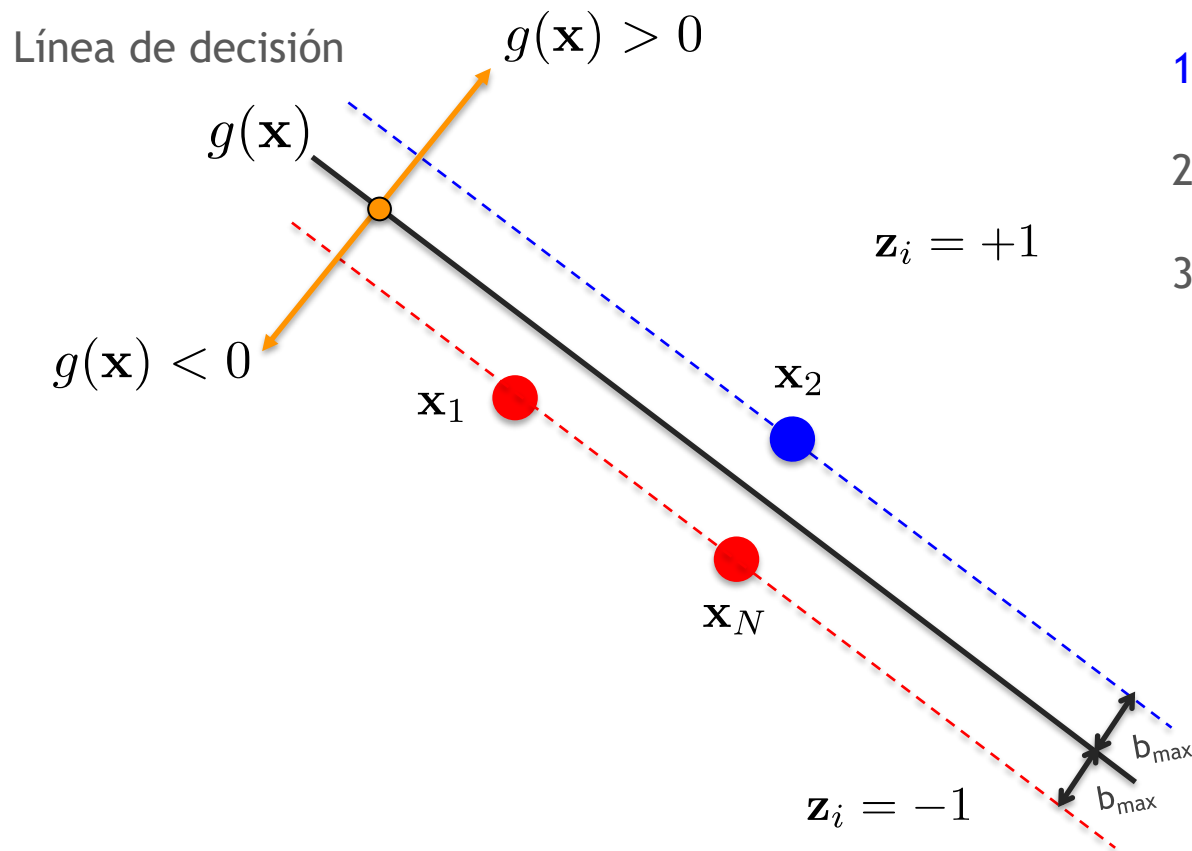
Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.



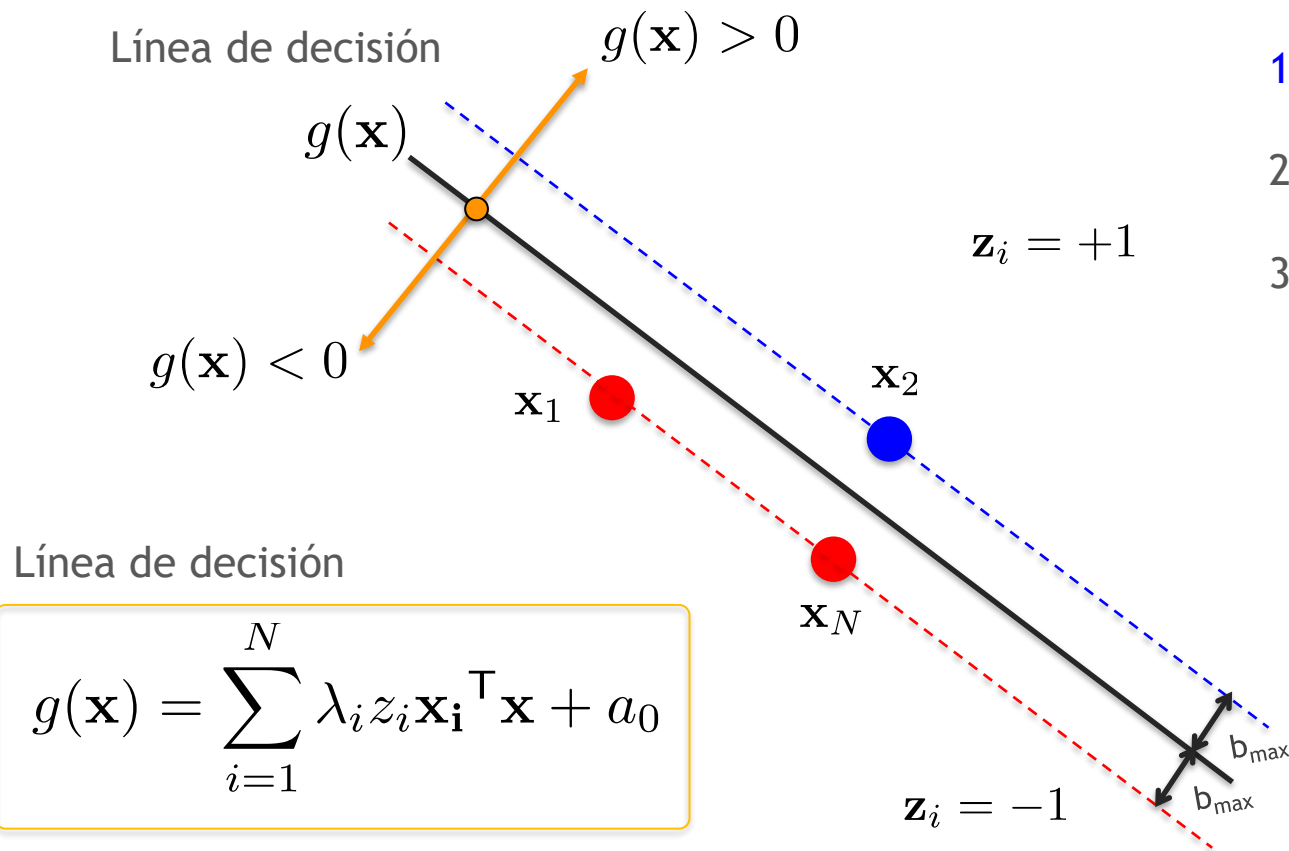
SVM: dos clases



Ideas clave de SVM:

- 1) $b_1 = b_2 = b$.
- 2) b debe maximizarse.
- 3) Solución: $g(\mathbf{x})$

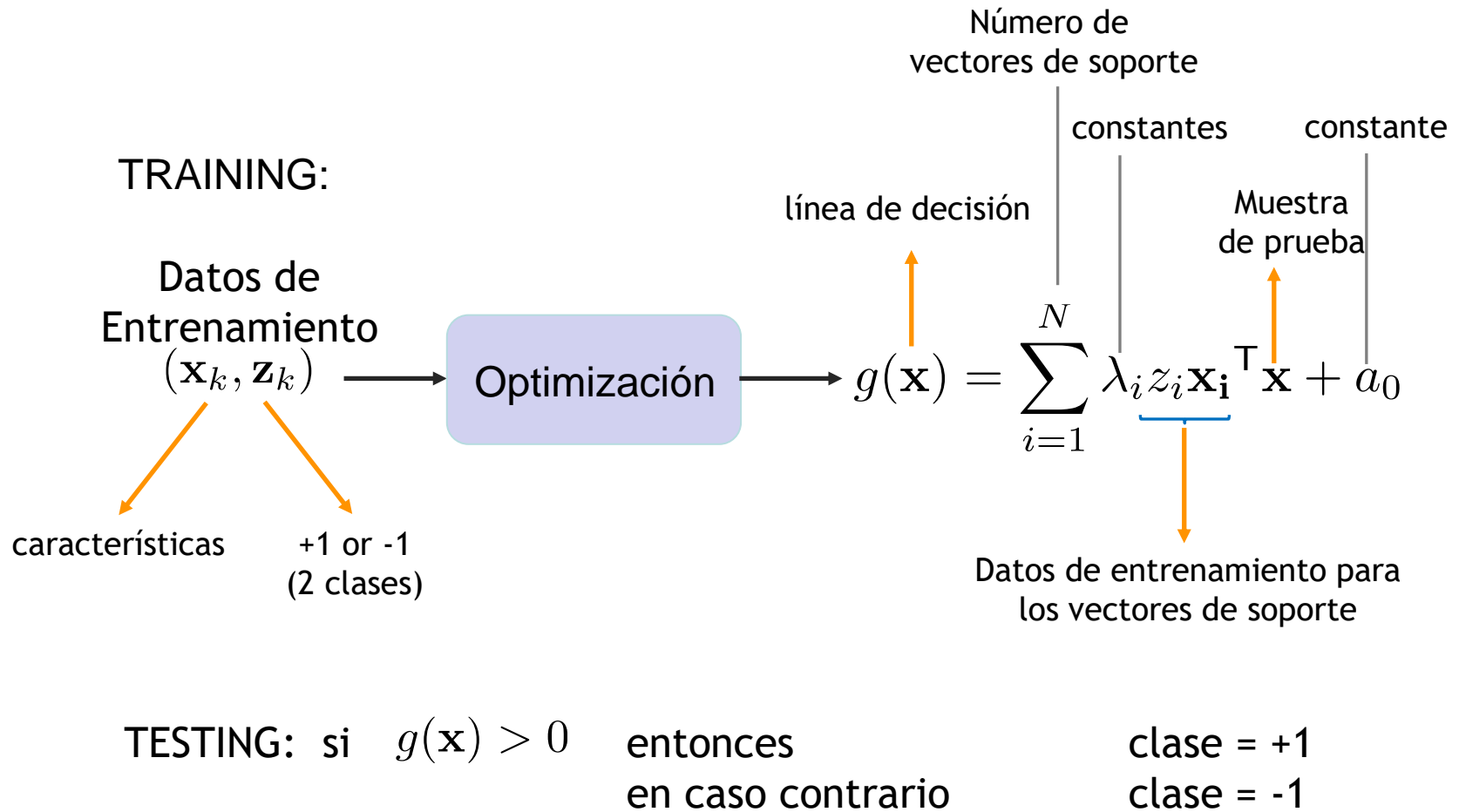
SVM: dos clases



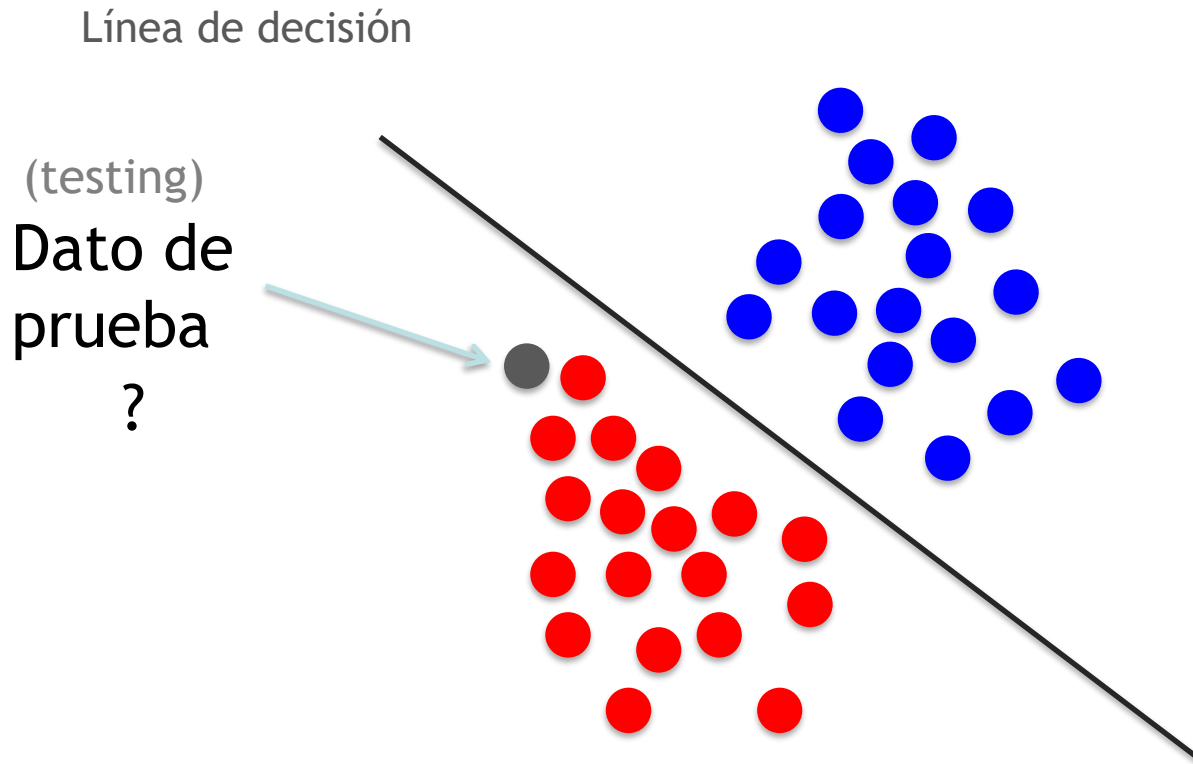
Ideas clave de SVM:

- 1) $b_1 = b_2 = b$.
- 2) b debe maximizarse.
- 3) Solución: $g(\mathbf{x})$

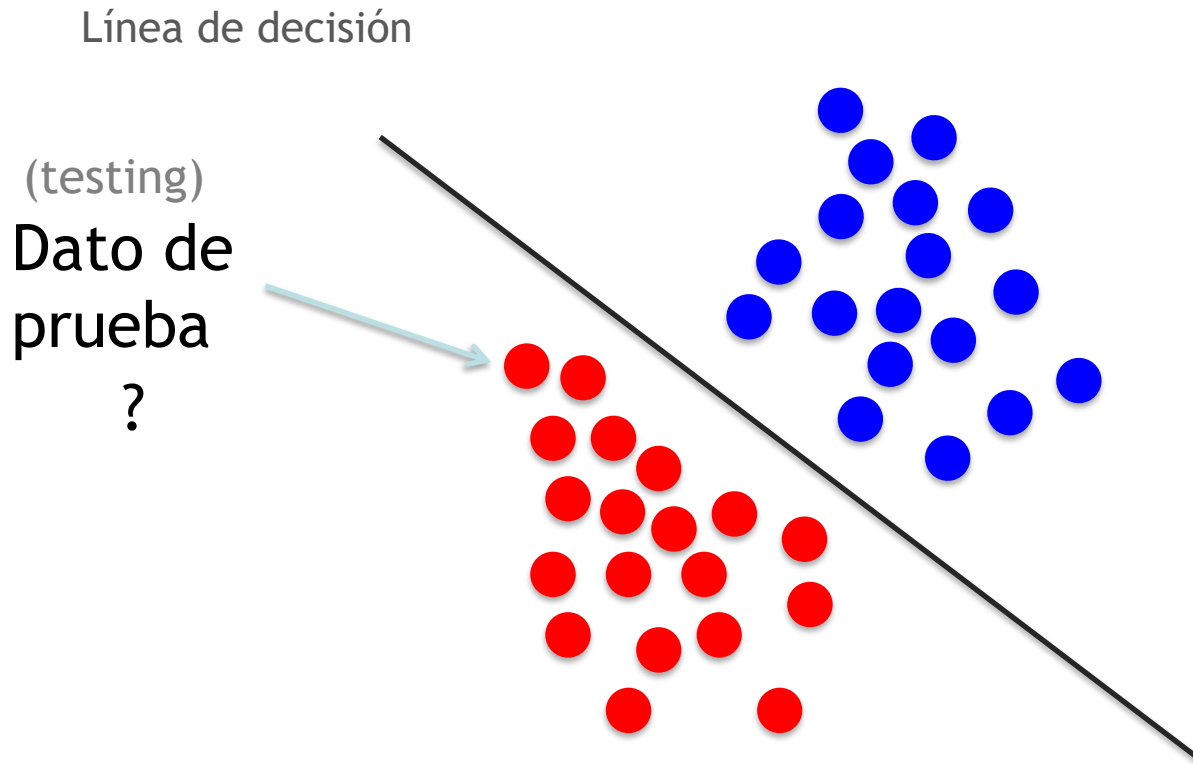
La solución de SVM



SVM: dos clases

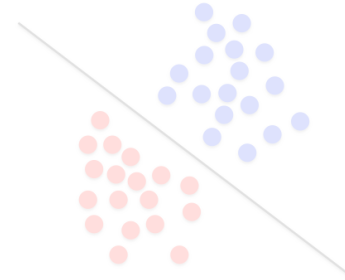


SVM: dos clases

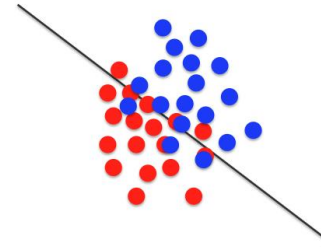


SVM: Máquinas vectoriales de soporte

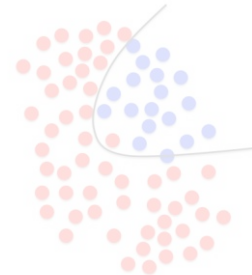
1) Lineal con separación perfecta



2) Lineal sin separación perfecta

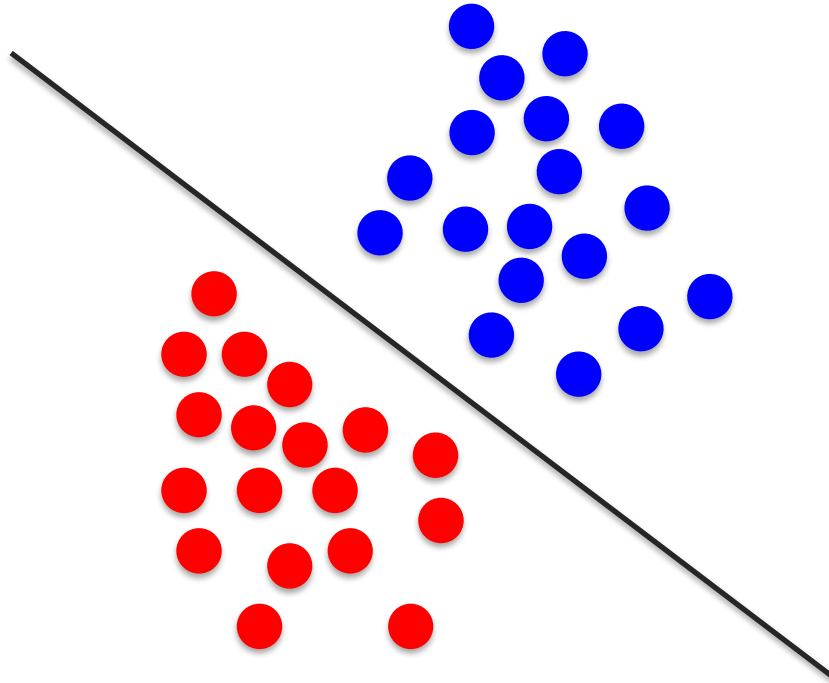


3) No lineal



SVM: dos clases

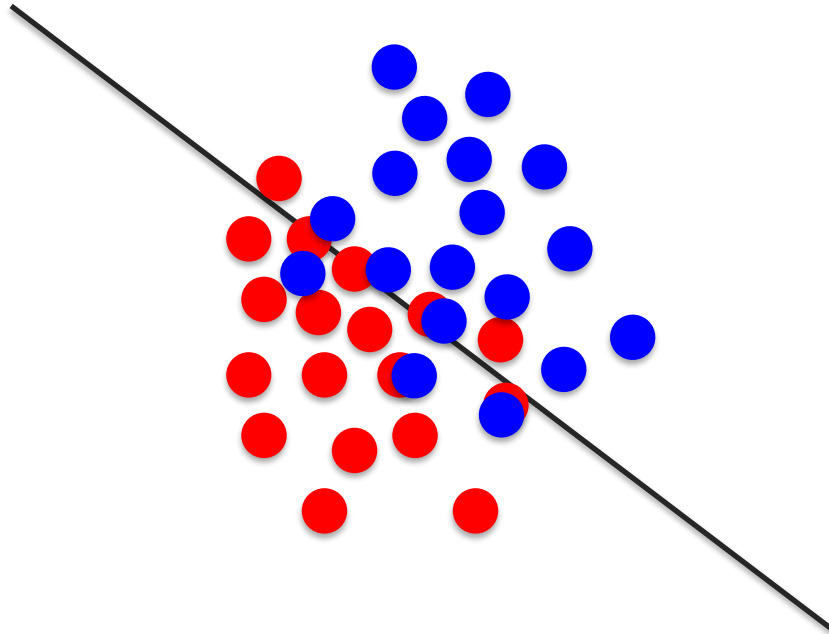
¿Cómo definir la línea de decisión cuando no hay una separación perfecta?



SVM: dos clases

¿Cómo definir la línea de decisión cuando no hay una separación perfecta?

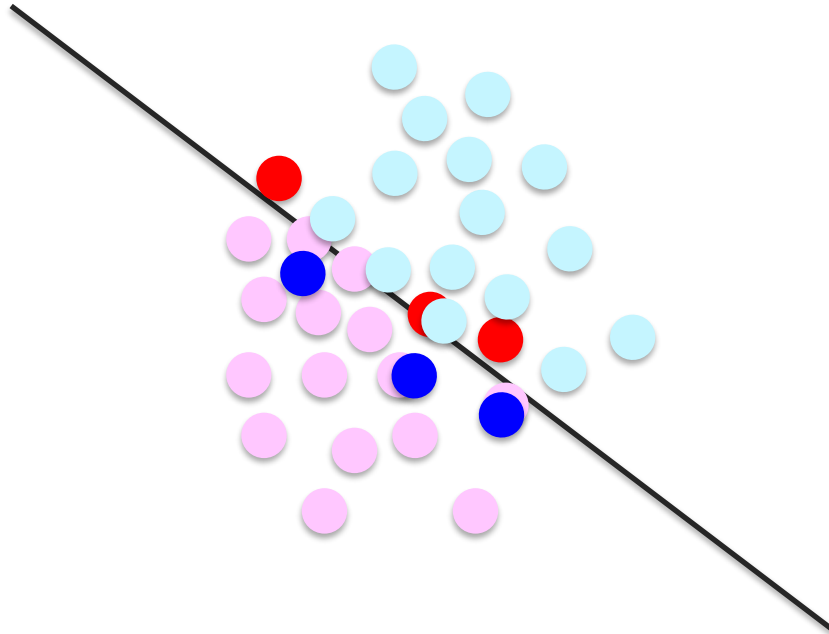
Consideramos sólo las muestras clasificadas erróneamente



SVM: dos clases

¿Cómo definir la línea de decisión cuando no hay una separación perfecta?

Consideramos sólo las muestras clasificadas erróneamente

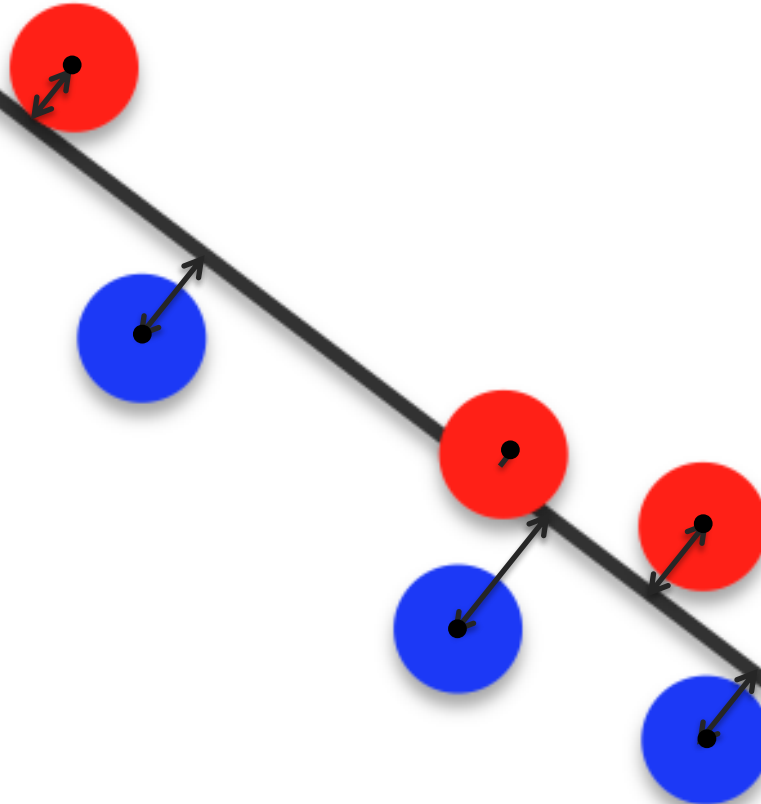


SVM

¿Cómo
decisi
separ

muestras
ente

Vectores de soporte



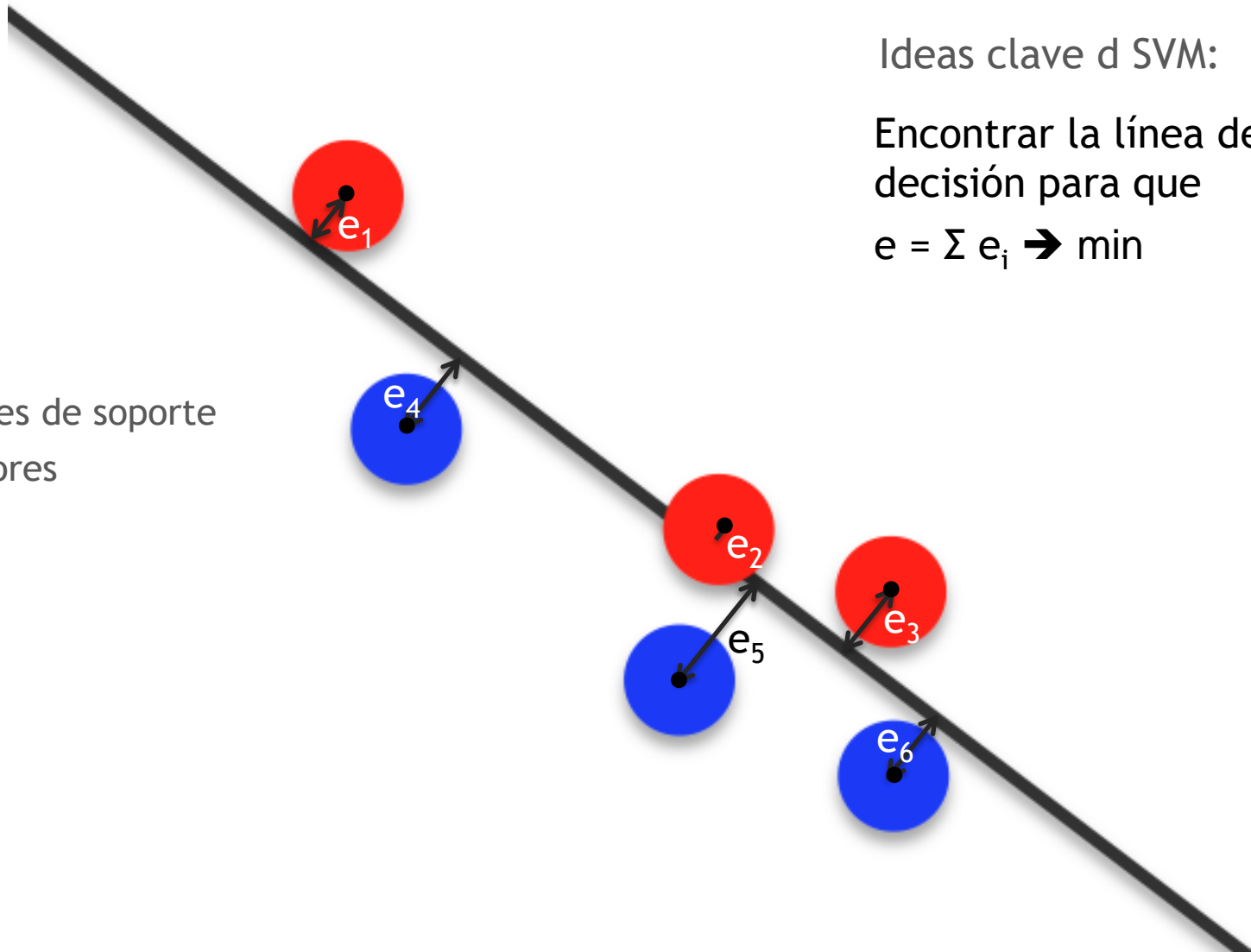
Ideas clave d SVM:

Encontrar la línea de
decisión para que

$$e = \sum e_i \rightarrow \min$$

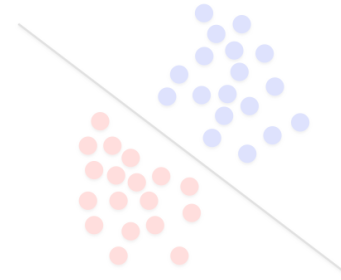
Vectores de soporte

e_i : errores

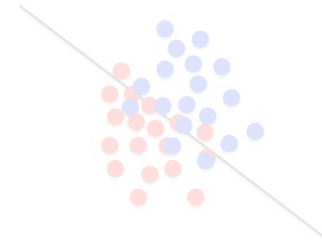


SVM: Máquinas vectoriales de soporte

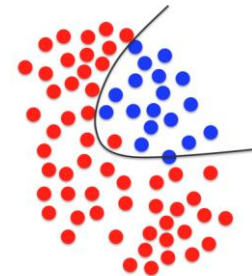
1) Lineal con separación perfecta

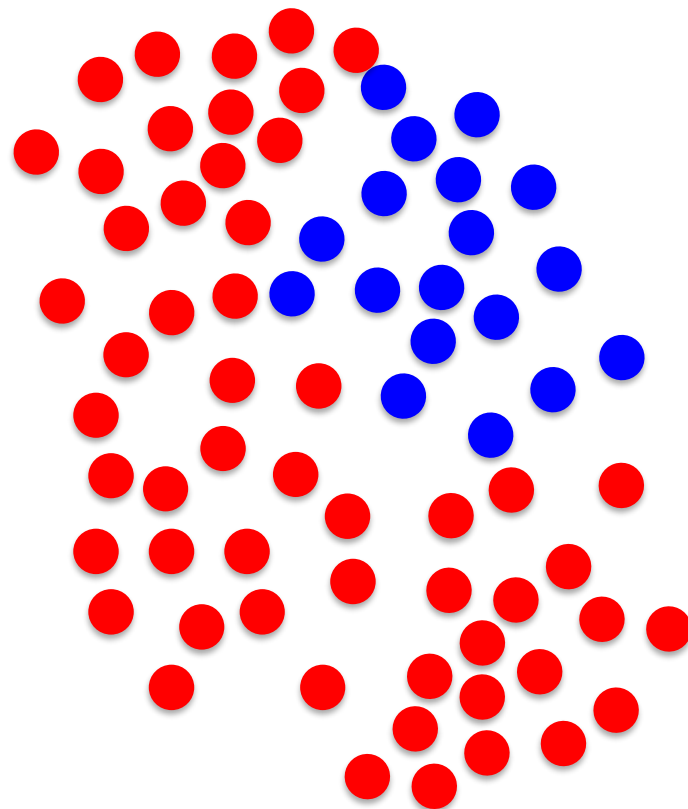


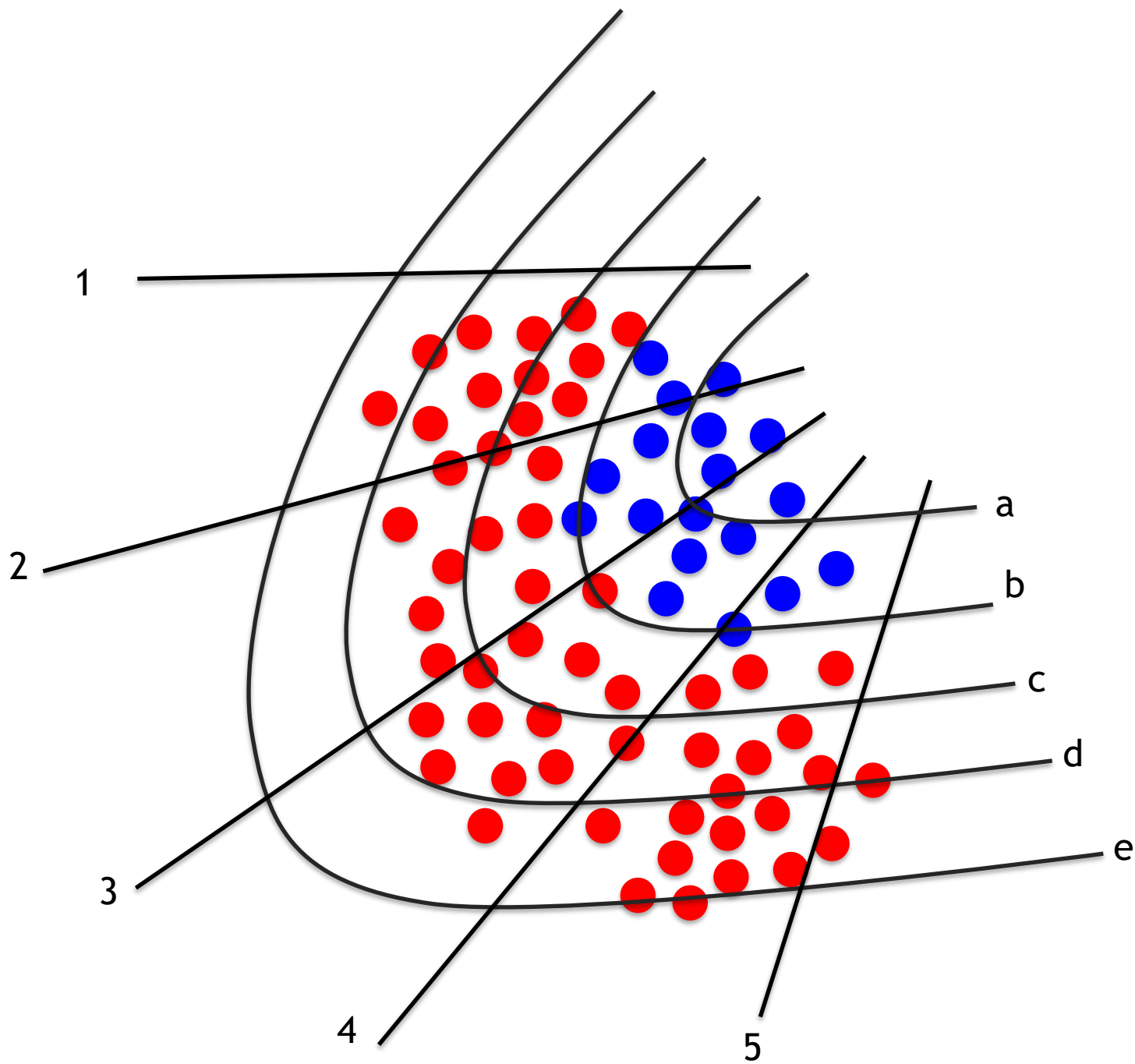
2) Lineal sin separación perfecta

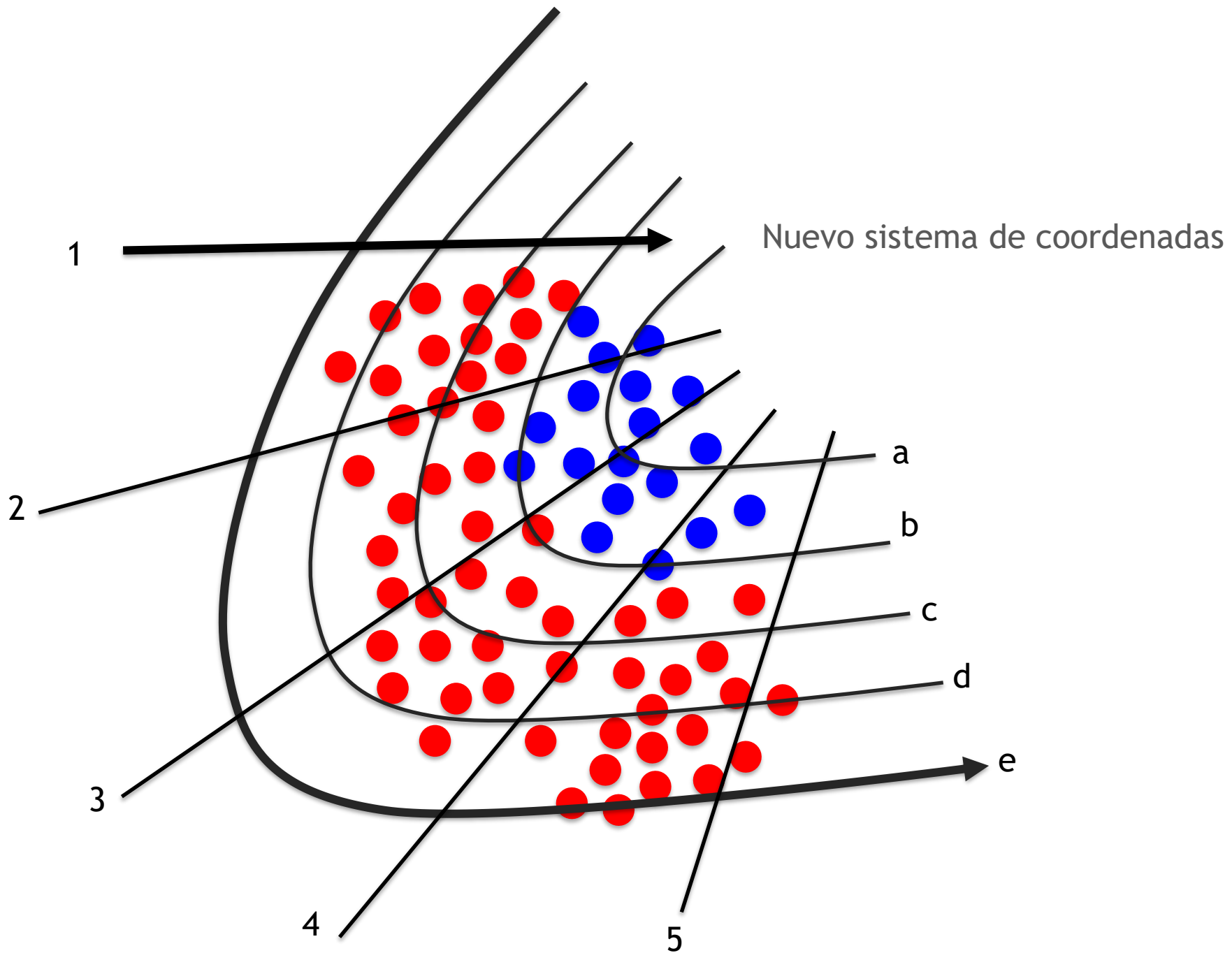


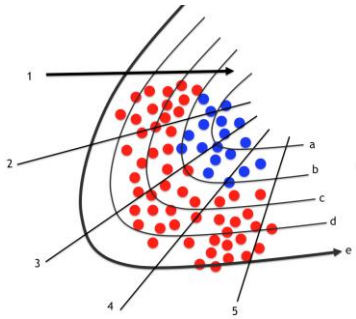
3) No lineal



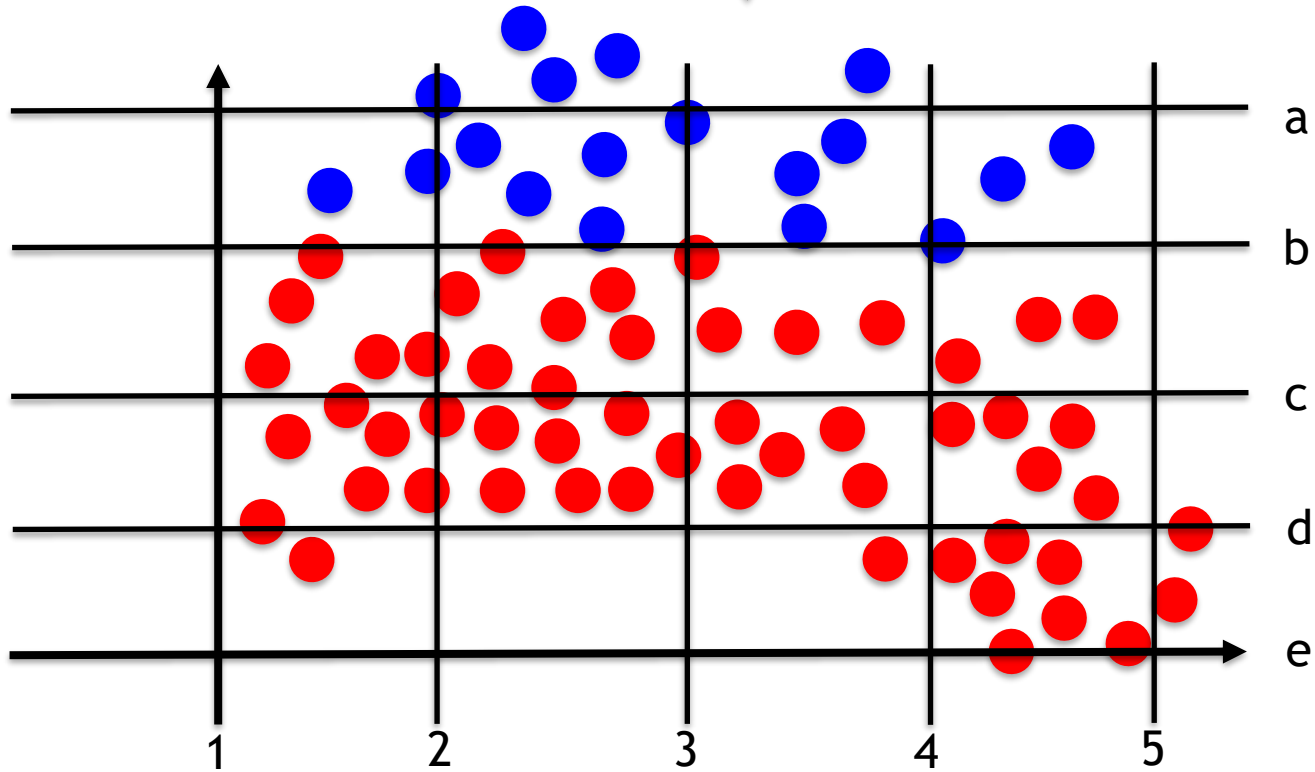


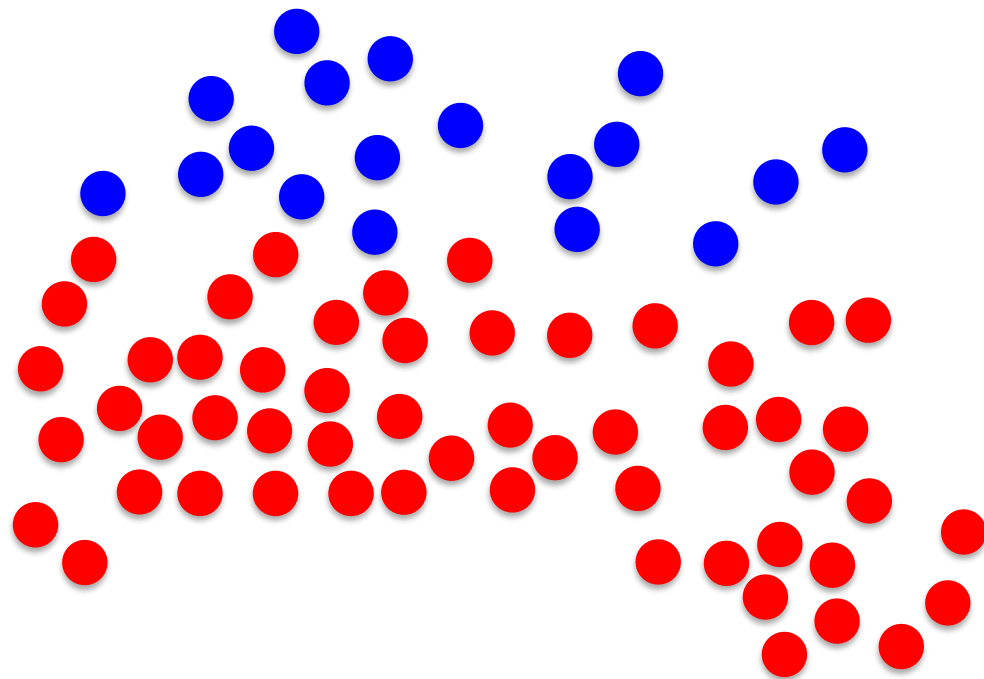




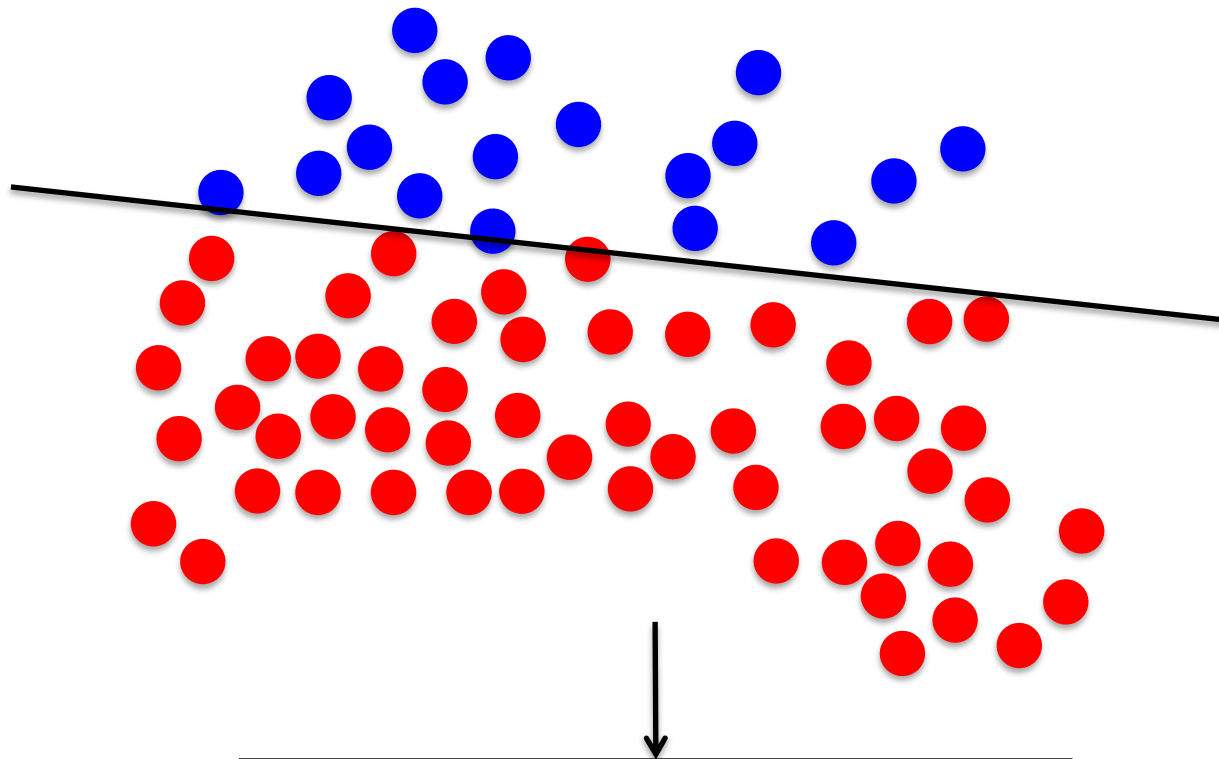


Transformación geométrica no lineal

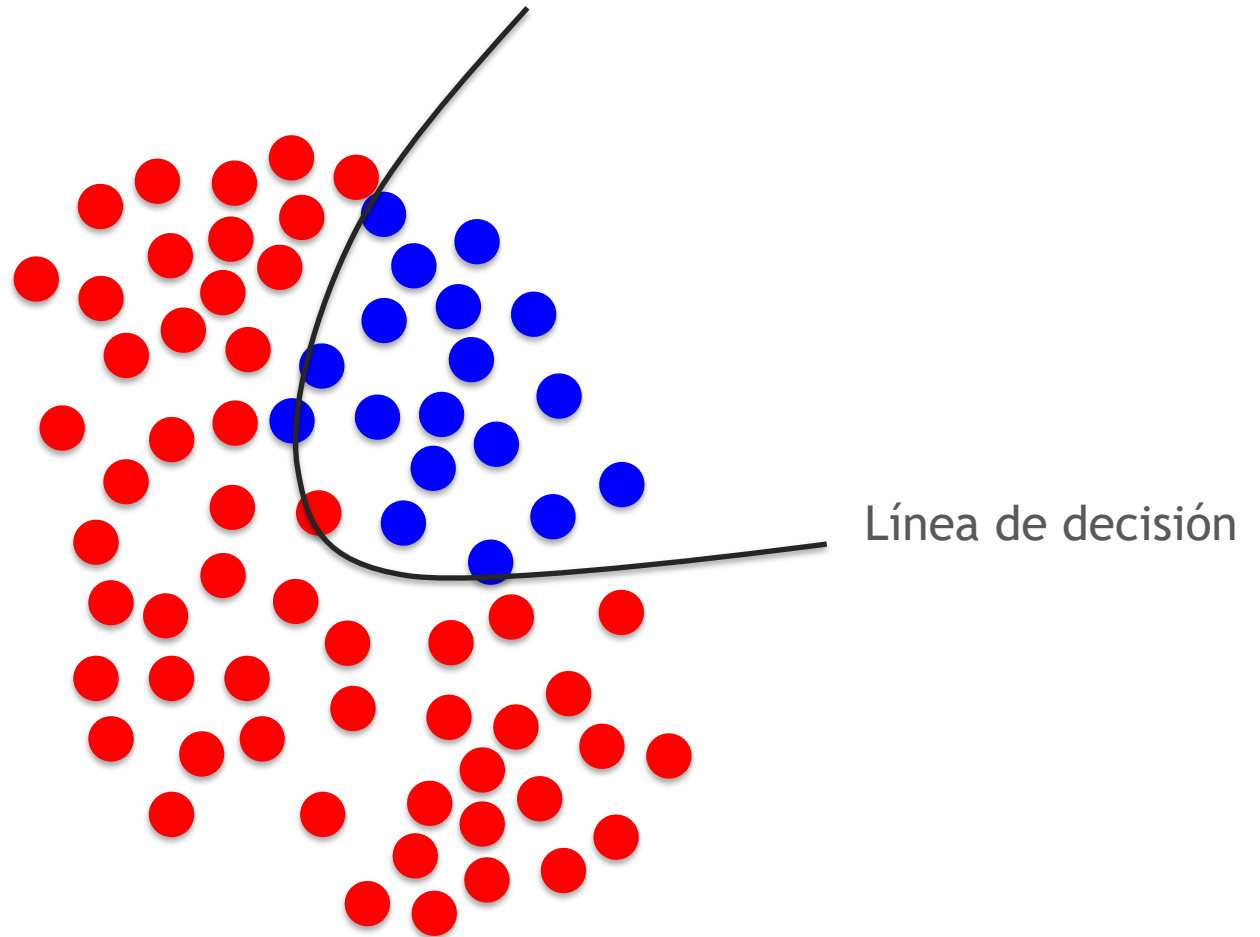




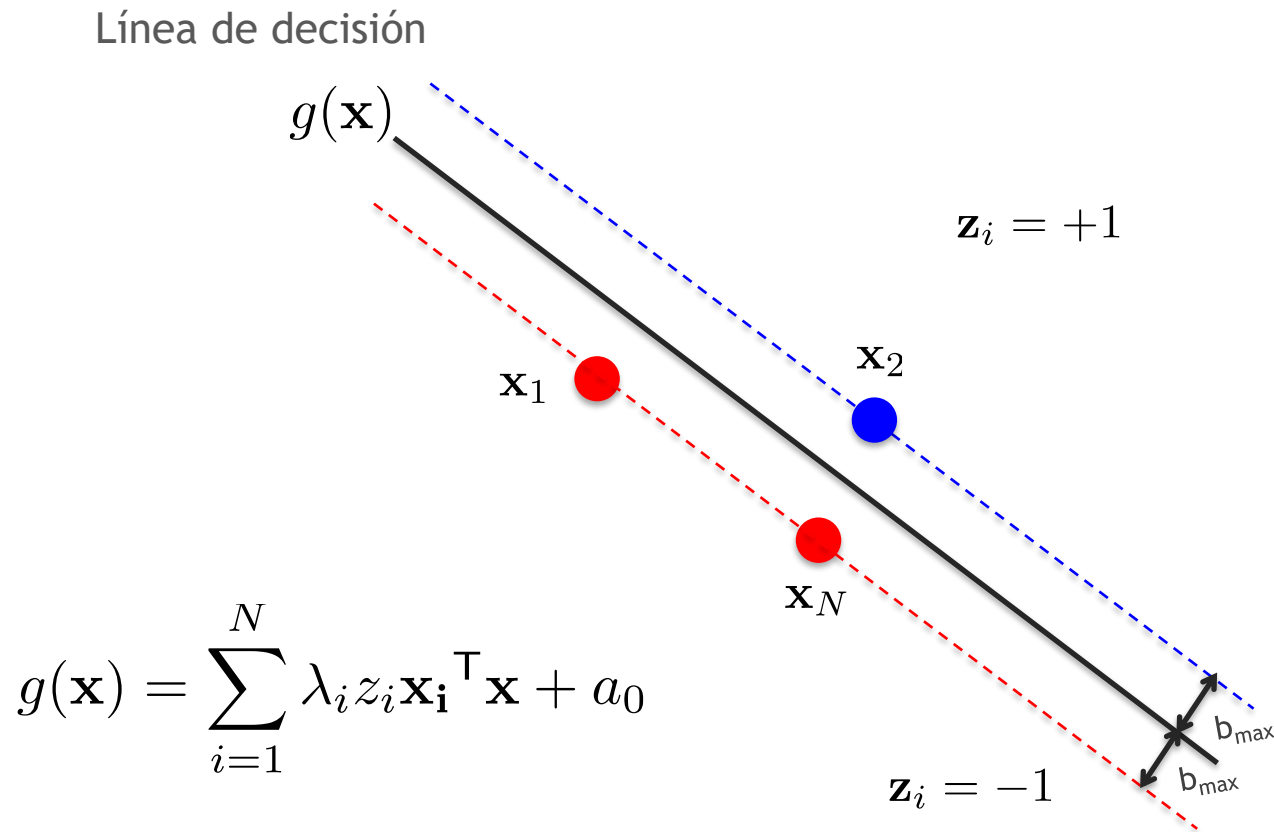
SVM lineal en nuevo sistema de coordenadas



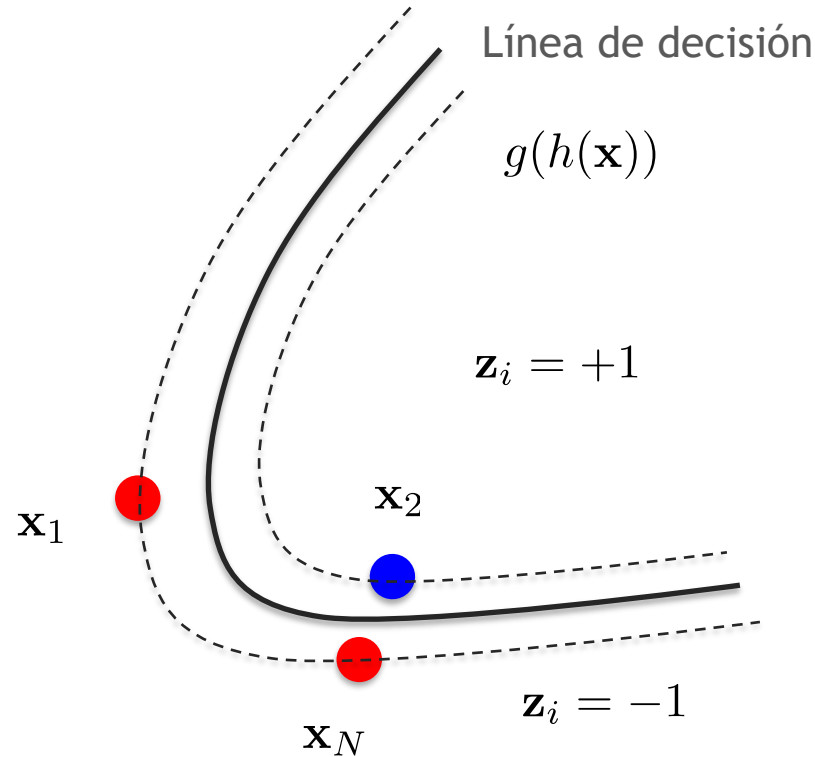
Transformación geométrica no lineal



SVM: Máquinas vectoriales de soporte



SVM: Máquinas vectoriales de soporte



$$g(h(\mathbf{x})) = \sum_{i=1}^N \lambda_i z_i h(\mathbf{x}_i)^T h(\mathbf{x}) + a_0$$

$$g(h(\mathbf{x})) = \sum_{i=1}^N \lambda_i z_i \langle h(\mathbf{x}_i), h(\mathbf{x}) \rangle + a_0$$

No se necesita $h(\mathbf{x})$, solo es necesario el kernel $\langle h(\mathbf{x}_i), h(\mathbf{x}) \rangle$

SVM: Kernels

$$K(\mathbf{x}', \mathbf{x}) = \langle h(\mathbf{x}'), h(\mathbf{x}) \rangle =$$

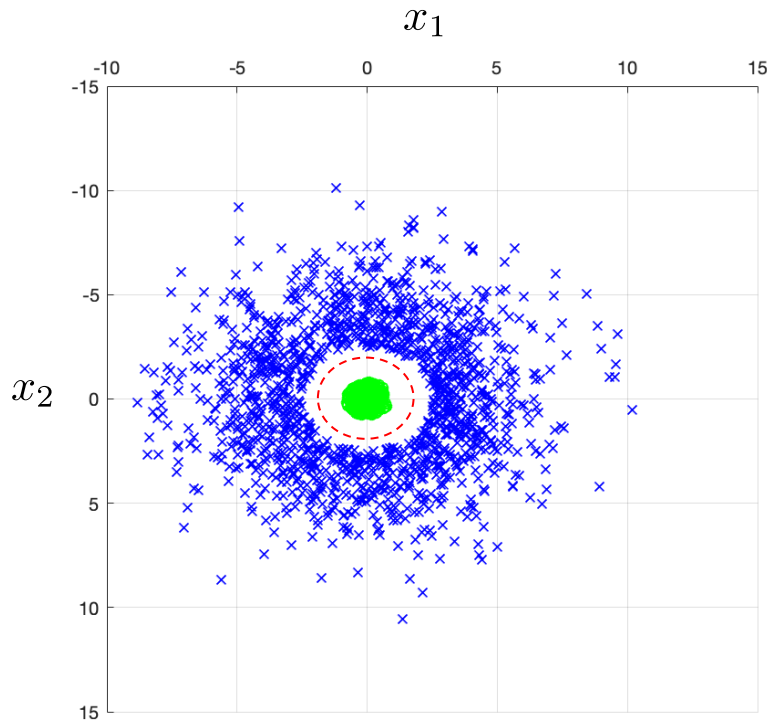
linear	$\langle \mathbf{x}', \mathbf{x} \rangle$
--------	---

polynomial	$(1 + \langle \mathbf{x}', \mathbf{x} \rangle)^n$
------------	---

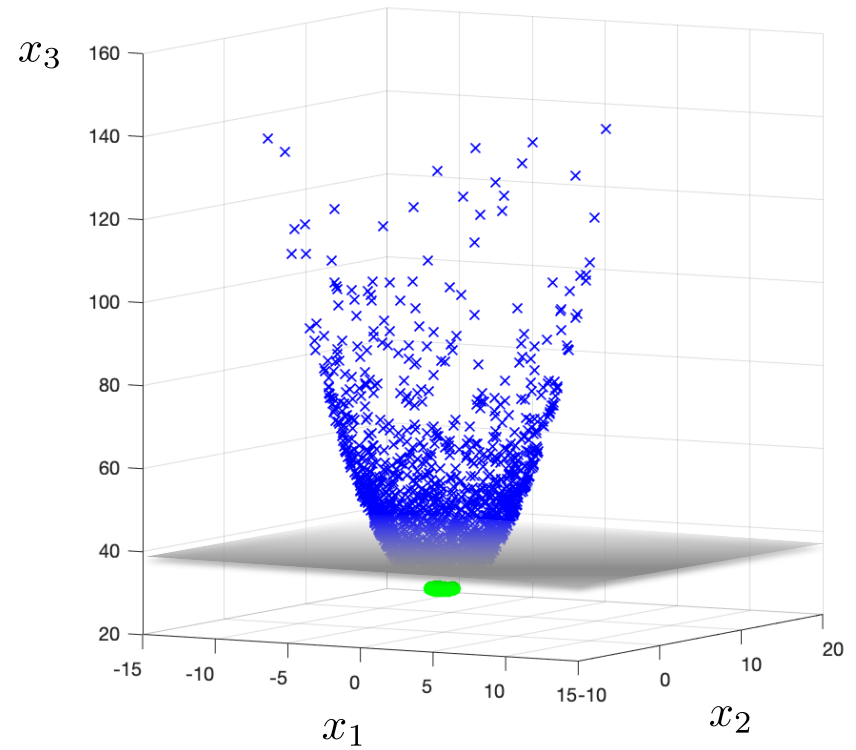
radial basis	$\exp(-\ \mathbf{x}' - \mathbf{x}\ ^2 / c)$
--------------	---

sigmoid	$\tanh(K_1 \langle \mathbf{x}', \mathbf{x} \rangle + K_2)$
---------	--

SVM: El truco del Kernel



La separación lineal es imposible

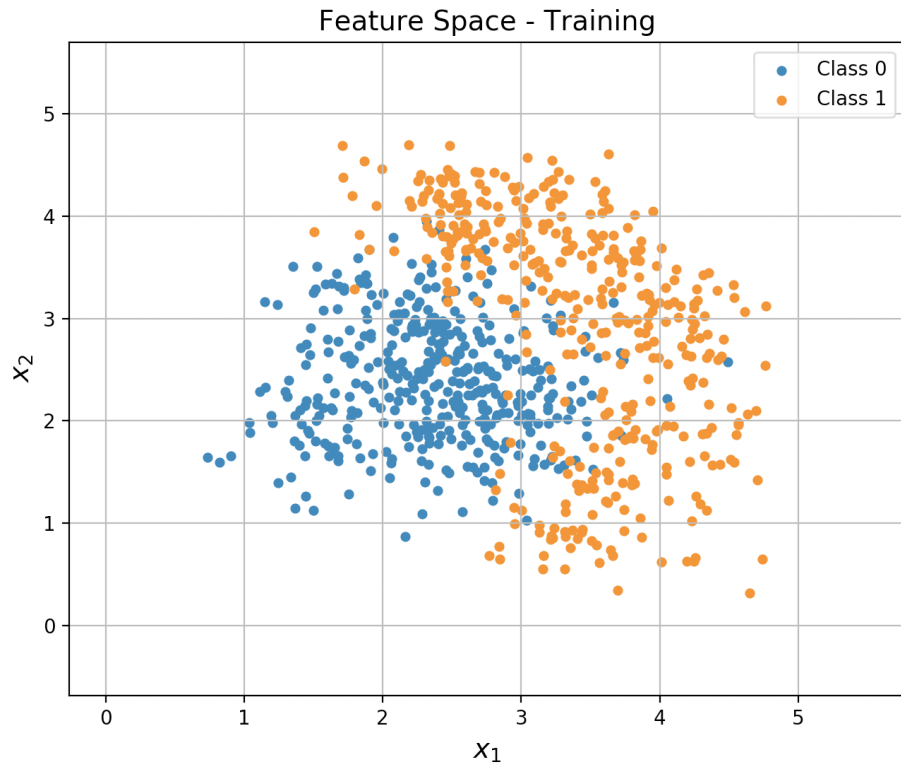


La separación lineal es perfecta

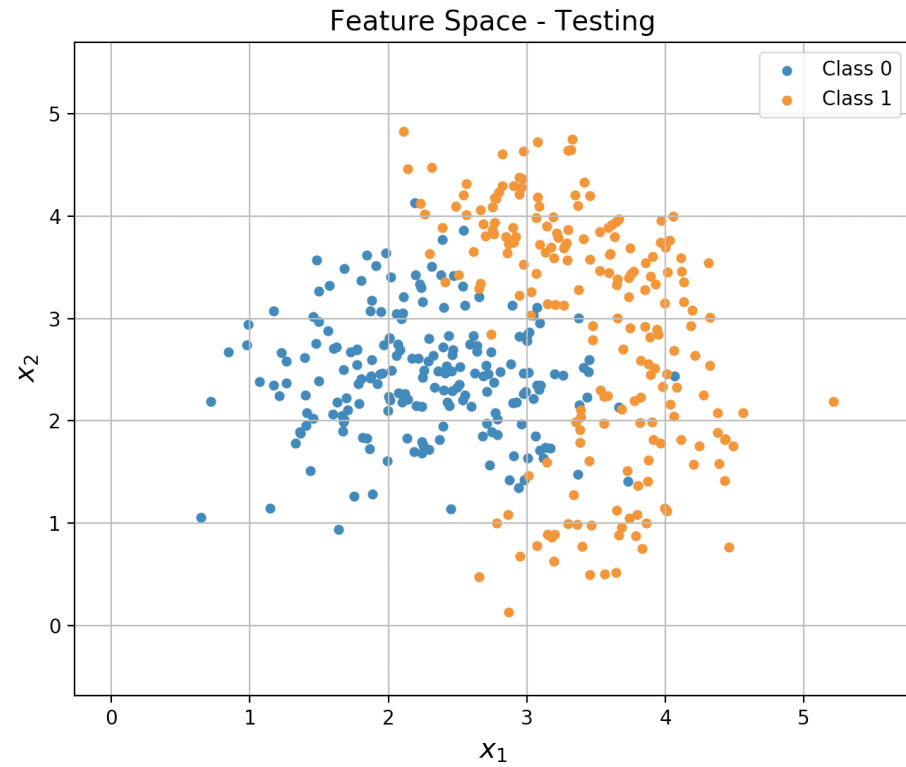
Ejemplos

Ejemplo

(X, d)
TRAINING



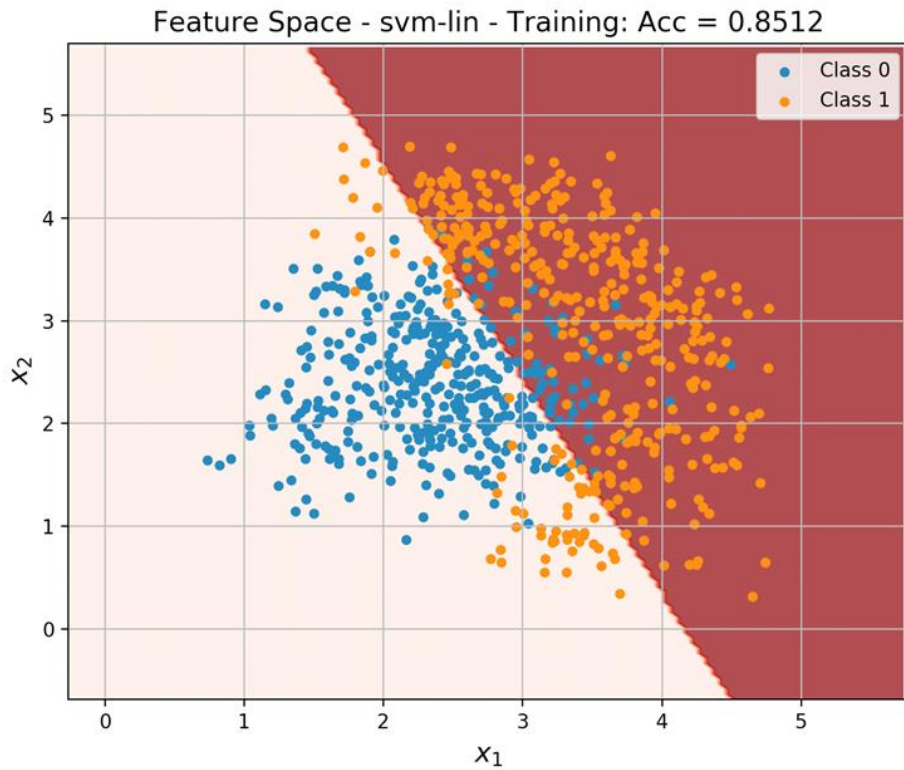
(X_t, d_t)
TESTING



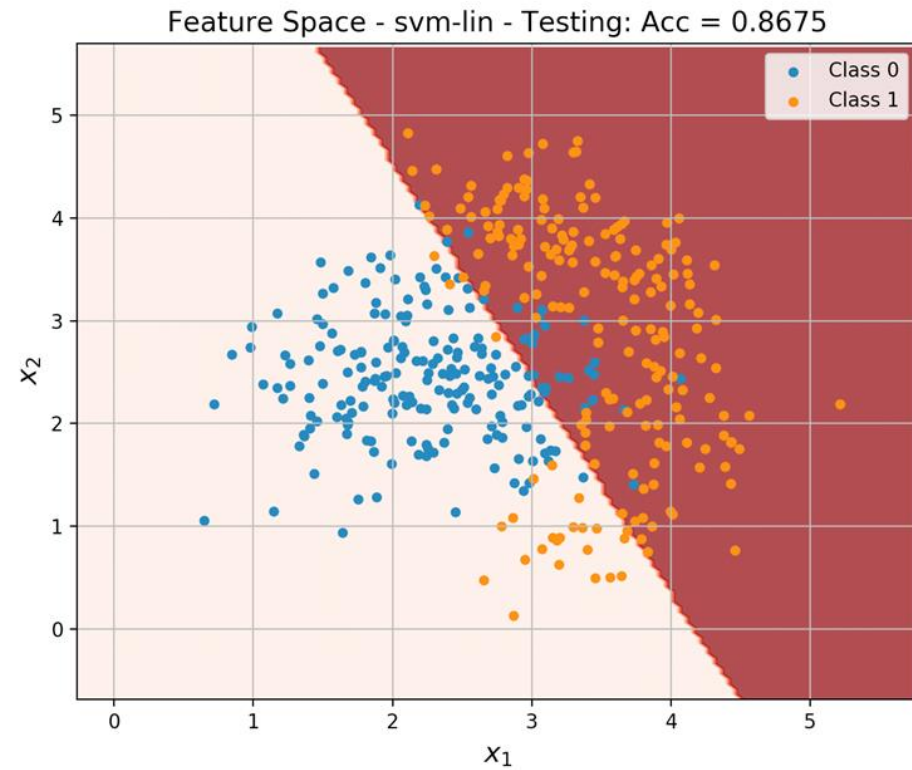
Ejemplo

SVM-LIN

(X, d)
TRAINING



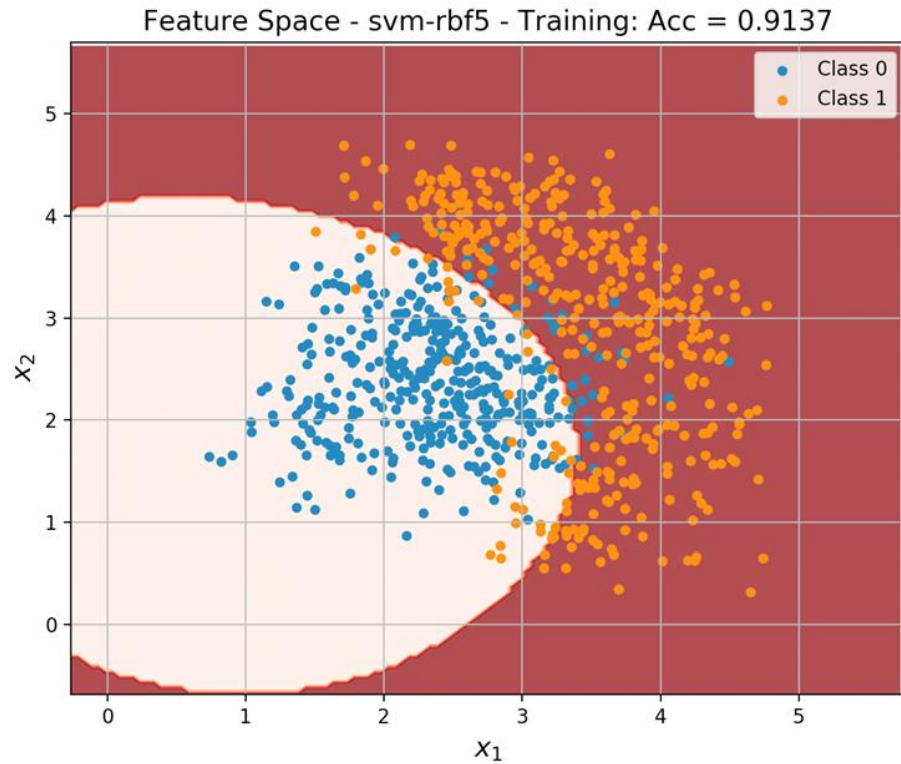
(X_t, d_t)
TESTING



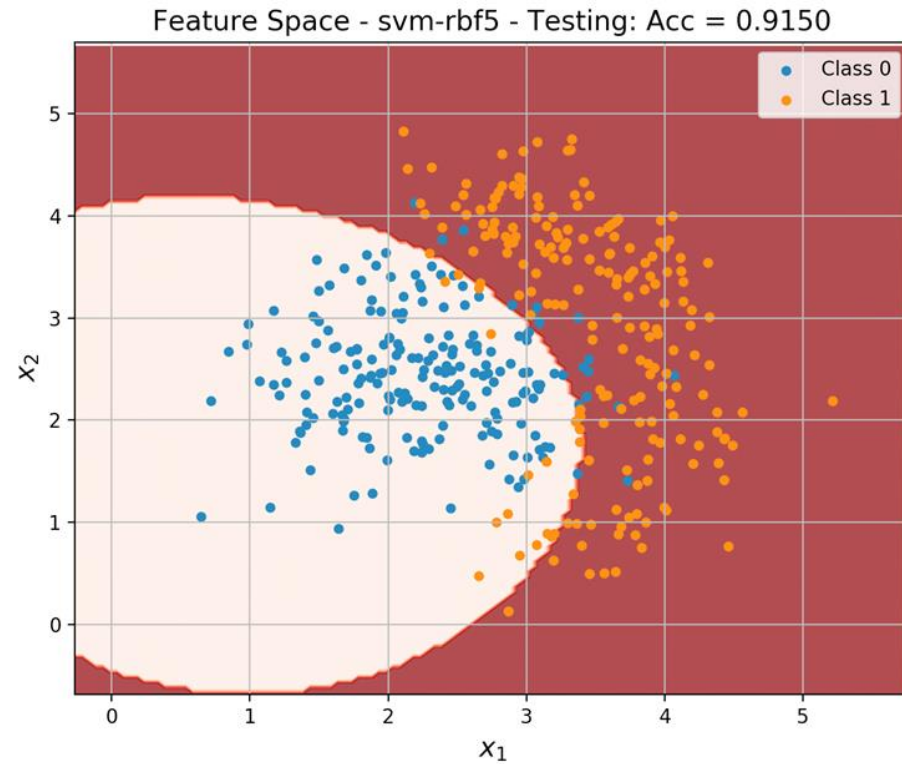
Ejemplo

SVM-RBF

(X, d)
TRAINING



(X_t, d_t)
TESTING



Bibliografía

- En memoria del profesor Patrick Winston. https://en.wikipedia.org/wiki/Patrick_Winston
- Corinna Cortes and V. Vapnik, "Support-Vector Networks", Machine Learning, 20, 1995. Disponible en <http://homepages.rpi.edu/~bennek/class/mml/papers/svn.pdf>
- Christopher J. C. Burges. "A Tutorial on Support Vector Machines for Pattern Recognition". Data Mining and Knowledge Discovery 2:121–167, 1998. Disponible en <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/svmtutorial.pdf>
- A Library for Support Vector Machines aplicación en línea <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- 16. Learning: Support Vector Machines <https://www.youtube.com/watch?v=PwhiWxHK8o>
- CVM Applet. aplicación en línea <http://www.eee.metu.edu.tr/~alatan/Courses/Demo/AppletSVM.html>
- Bases para el código de ejemplo en Opencv tomado de <http://opencv.jp/sample/svm.html> “en japonés”
- Support Vector Machine Disponible en http://en.wikipedia.org/wiki/Support_vector_machine
- SVM with polynomial kernel visualization disponible en <http://www.youtube.com/watch?v=3liCbRZPrZA>
- Excelente video resumiendo el concepto: <https://www.youtube.com/watch?v=efR1C6CvhmE>